NASA CONTRACTOR REPORT

NASA CR-1729

AN ALGORITHM FOR LIAPUNOV STABILITY ANALYSIS OF COMPLEX NONLINEAR SYSTEMS WITH APPLICATION TO THE ORBITING ASTRONOMICAL OBSERVATORY

by Gunther R. Geiss, Victor D. Cohen, Robert D'heedene, David Rothschild, and Arthur Chomas

Prepared by
GRUMMAN AEROSPACE CORPORATION
Bethpage, N. Y. 11714
for Ames Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • MAY 1971

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-1729		

4. Title and Subtitle	5. Report Date
"An Algorithm for Liapunov Stability Analysis of Complex Nonlinear Systems With Application to the Orbiting Astronomical Observatory"	May 1971
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
Gunther R. Geiss, Victor D. Cohen, Robert D'heedene, D. Rothschild and Arthur Chomas	
	10. Work Unit No.
9. Performing Organization Name and Address	11. Contract or Grant No.
Grumman Aerospace Corporation Bethpage, New York	NAS 2-4063
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Contractor Report
National Aeronautics & Space Administration Washington, D.C. 20546	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

The report describes the development of an algorithm for estimating the domain of attraction of equilibrium of a "paired tracker" coarse attitude control mode for the orbiting Astronomical Observatory. The algorithm generates an "optimal" quadratic Liapunov function by formulating the domain estimate as a min-max problem solved by random search techniques. The importance of the choice of state variables for the model is emphasized. Testing the algorithm on the full nine-dimensional model and a six-dimensional approximation gave comparable results. Though the estimated domain of attraction is small in comparison with direct stimulation, the region extends well beyond inception of the dominant saturating nonlinearity.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
stability Liapunov functions nonlinear system stability min-max problem optimization random search	Unclassified-Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	181	3.00

*For sale by the National Technical Information Service, Springfield, Virginia 22151

FOREWORD

The reported research was performed by the Grumman Aerospace Corporation Research Department and was supported by the National Aeronautics and Space Administration. Mr. Brian Doolin, Assistant Chief, Theoretical Guidance and Control Branch, NASA Ames Research Center, was the NASA technical representative for the study reported here. Prior research had been supported by the NASA George C. Marshall Space Flight Center, Huntsville, Alabama, with Mr. Commodore C. Dearman, Jr., as the NASA technical representative.

Dr. Gunther Geiss was the principal investigator throughout the study. During basic research on the algorithm he was aided by Dr. John V. Abbate, and Messrs. James Alberi, Dushan Boyanovitch, Robert McGill, David Rothschild, and Gerald Taylor. Studies directed specifically at stability of the OAO coarse pointing mode involved Drs. Victor Cohen and Robert D'heedene, and Messrs. Art Chomas, W. Feldman, and David Rothschild.

The problem was brought to our attention by Mr. Roy Olsen of the Grumman OAO project. Support from Messrs. Roy Olsen, Erik Eriksen, and Robert Rennie of the Grumman OAO project is gratefully acknowledged.

Messrs. Brian Doolin and C. C. Dearman, Jr., of NASA provided support and encouragement during various phases of the study, for which we are grateful. We thank Dr. George Meyer and Mr. Robert D. Showman of the NASA Ames Research Center who provided constructive and detailed technical criticism. The help and support of Mr. W. Hibbard and Dr. W. Redisch of the NASA Goddard Space Flight Center in the early phases of the study were valuable. The authors are grateful to Dr. John F. Clark, Director, Goddard Space Flight Center, and Dr. A. Levine, Executive Director, Goddard Institute for Space Studies, for making the IBM 360/95 facility at the Institute available to us. Dr. David L. Stonehill of Computer Applications, Inc. contributed significantly in adapting our program to the systems implemented on the IBM 360/95.

ABSTRACT

An algorithm is described for automatic computation of a quadratic estimate of the domain of stability for the stable equilibrium states of nonlinear systems of ordinary differential equations. The study was motivated by the failure of standard linear stability analysis techniques to predict adequately the stability of the current NASA Orbiting Astronomical Observatory (OAO) coarse pointing mode control system. Since a new version of this particular control system was used for the primary feasibility test of the algorithm, modeling and simulation results for this control system are reported. In developing the algorithm various minimization and random search techniques were utilized to solve the min-max problem which yields the estimate; the result of the reported experimentation and evaluation was rejection of gradient search and penalty function techniques as being inapplicable to this particular problem and high order nonlinear problems in general. The new methods developed and described here are the first known example of solving a min-max problem via two random searches. The algorithm has been extensively tested, and although apparently expensive in machine time it is potentially more cost-effective than simulation, which is the only competitive technique for high order systems. Noteworthy is the fact that methods used by several authors for problems of dimension $n = 2$, 3, and 4 and that are claimed to "generalize easily" to higher dimensions are not feasible for complicated physical systems with $n = 6$ or $n = 9$, the practical cases considered here.

The modeling and simulation studies show that the choice of sensor (star tracker) model has a significant bearing on the difficulty of the stability analysis, and that neither the linear approximation nor the Popov approximation (linear part plus saturations) adequately represents the system stability properties. Thus the numerical algorithm developed is necessary for the stability analysis of this system.

In examining the use of Lurć-Liapunov functions and perturbation techniques to obtain an improved estimate it is shown that the Popov approximation cannot be analyzed completely with the available frequency domain techniques and that computational aids are required to effectively apply frequency domain techniques to this complex physical problem.

The problems of numerically solving the Liapunov matrix equation and generating arbitrary positive definite matrices were solved in the development of the algorithm.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Cont)

LIST OF ILLUSTRATIONS (Cont)

LIST OF TABLES

LIST OF SYMBOLS

A	matrix of linear part of state equation
A_{mod}	$n^2 \times n^2$ matrix formed from A
a_{11}, a_{13}	design constants chosen to be zero
C_i	constant
$\mathcal{B}(0)$	domain of attraction of the origin
d_{12}	design constant
det	determinant
$\text{diag}\{\lambda_1, \ldots, \lambda_n\}$	diagonal matrix with elements λ_1 through λ_n
$F(x)$	r.h.s. of state differential equation
$\frac{\partial F}{\partial x}\Big\|_{x=x_e}$	matrix of linear part of state equation
$f(V_i')$	motor saturation function
$G(s)$	transfer function
$g(x)$	nonlinear part of state equation
H_{ω_ϕ}, H_{ω_θ}, H_{ω_ψ}	roll, pitch, and yaw momentum
Ih_ϕ^o, Ih_θ^o, Ih_ψ^o	initial total roll, pitch, and yaw momenta
I	principal moment of inertia
I_n	$n \times n$ identity matrix
I_{ij}	element of inertia tensor
$J(Q)$	objective function, proportional to the volume of the domain of attraction

K	limiter constant (100 volts)
K_C	compensator gain (2.685×10^5 volts/rad)
K_m	motor gain (1/13 ft-lb-sec/volt)
k_1, k_2	weighting constants in penalty function formulation
ℓ	minimum value of the Liapunov function on the surface where the total time derivative of the function is zero
P	positive definite symmetric matrix
p_{ij}	elements of P
p, q, r	roll, pitch, and yaw body rates
Q	positive definite symmetric matrix
Q^o	optimal Q
q_{ij}	elements of Q
$Re(\cdot)$	real part of $(\cdot)$
R_α, R_β, R_γ	rotation transformations
R_ϕ, R_θ, R_ψ	rotation transformations
S	orthogonal matrix
sat $(\cdot)$	saturation function
sgn $(\cdot)$	"sign" or signum function
T	rotation matrix
$V(x)$, V	Liapunov function
V_i	value of V
$\dot{V}(x) = \dot{V}$	total time derivative of Liapunov function along the trajectories of the differential equation for x
V'_ϕ, V'_θ, V'_ψ	compensator output voltage
V''_ϕ, V''_θ, V''_ψ	saturated compensator voltage outputs

v_ϕ, v_θ, v_ψ	roll, pitch, and yaw wheel momenta
v'_ϕ, v'_θ, v'_ψ	redefined (zero equilibrium) wheel momentum components
v''_ϕ, v''_θ, v''_ψ	nondimensionalized (zero equilibrium) wheel momentum components
x	state vector
x_e	equilibrium state
x'	redefined state with zero equilibrium
x''	nondimensionalized redefined state with zero equilibrium
x_o	initial state
$x(t; x_o)$	state trajectory beginning at x_o for $t = 0$
X_b, Y_b, Z_b	axes of body coordinate frame
X_r, Y_r, Z_r	axes of inertial reference coordinate frame
X_T, Y_T, Z_T	axes of tracker coordinate frame
X_{TR}, Y_{TR}, Z_{TR}	axes of tracker reference coordinate frame
β_{1C}, β_{2C}	commanded inner gimbal angles for trackers 1 and 2
γ_{1C}, γ_{2C}	commanded outer gimbal angles for trackers 1 and 2
$\Delta\beta_i$, $\Delta\gamma_i$	gimbal angle errors
$\Delta\beta_1^e$, $\Delta\beta_2^e$	equilibrium values of inner gimbal angle errors
ϵ_ϕ, ϵ_θ, ϵ_ψ	roll, pitch, and yaw error signals
ϕ, θ, ψ	roll, pitch, and yaw Euler angles
ϕ_e, θ_e, ψ_e	roll, pitch, and yaw Euler angle equilibrium values
ϕ', θ', ψ'	redefined (zero offset) roll, pitch, and yaw Euler angles

Λ	diagonal matrix of eigenvalues
λ_i	eigenvalue
$\rho(x)$	penalty function term for nearing origin
$\tau_1 = 4.5$ sec $\tau_2 = 0.5$ sec	compensation time constants
$\tau_m = 76.8$ sec	motor time constant
Ω_ℓ	estimate of domain of attraction
ω_ϕ, ω_θ, ω_ψ	roll, pitch, and yaw compensator voltage state variables
ω'_ϕ, ω'_θ, ω'_ψ	ω_ϕ, ω_θ, and ω_ψ redefined to have zero equilibrium values
ω''_ϕ, ω''_θ, ω''_ψ	dimensionless ω_ϕ, ω_θ, and ω_ψ with zero equilibrium values
$\textcircled{x}$	Kronecker product
$\lvert\cdot\rvert$	absolute value of $(\cdot)$
$\lVert\cdot\rVert$	Euclidean norm of $(\cdot)$
[j]	refers to reference j
$(\dot{\cdot})$	time derivative
$\sim$	approximate or asymptotic equality
$\propto$	proportionality
{r|statement}	set of all r such that the statement is true
$(\ )^T$	transpose
c(•)	cosine (•)
s(•)	sine (•)
t(•)	tangent (•)
$(\tilde{\ })$	unit vector

AN ALGORITHM FOR LIAPUNOV STABILITY ANALYSIS OF COMPLEX NONLINEAR SYSTEMS WITH APPLICATION TO THE ORBITING ASTRONOMICAL OBSERVATORY

Gunther R. Geiss,† Victor D. Cohen, Robert D'heedene, David Rothschild, and Arthur Chomas

Grumman Aerospace Corporation
Bethpage, New York 11714

1. SUMMARY

This report describes the development of a numerical algorithm for determining the stability of nonlinear systems and its application to the "paired-tracker" attitude control system that has been proposed for the NASA Orbiting Astronomical Observatory (OAO). The objectives of the study were: 1) to demonstrate the feasibility of the algorithm in a nonacademic setting; and 2) to provide a tool for use in the analysis/design and operational parts of the OAO program. The study was motivated by the failure of standard linear analysis to adequately predict the stability of the present OAO attitude control system. Prior research indicated that the algorithm would provide results with higher confidence levels than simulation, which was the principal tool being used.

The algorithm is based on Liapunov stability theory and is applicable to systems that are described by a set of quasilinear differential equations of the form

$$\dot{x} = Ax + g(x) \qquad (1\text{-}1)$$

where x is the n vector of state variables, A is a stable matrix, $g(x)$ is at least of order x^2, and $x \equiv 0$ is the equilibrium point of interest [i.e., $g(0) = 0$]. This admits most systems that are designed by techniques currently used. The objective is to determine or estimate the set of initial states from

† Presently with Poseidon Scientific Corporation, Hauppauge, N.Y. 11787

which the system will return to the equilibrium state $x = 0$, i.e., to estimate the domain of attraction of the equilibrium state.

The objective is achieved by using LaSalle's theorem on the extent of asymptotic stability. Basically, the theorem states that if a Liapunov function $V(x)$ and its total time derivative have certain properties within the set Ω_ℓ of states x such that $V(x) < \ell$, where ℓ is a constant, then all trajectories beginning in Ω_ℓ tend toward the equilibrium $x = 0$. The problem of finding such a Liapunov function $V(x)$ is resolved by restricting consideration to positive definite quadratic forms. The largest value of ℓ for which the required conditions hold gives the best estimate relative to that particular quadratic form. This value is found to be the solution of a constrained minimum problem. To eliminate the dependence of the quality of the estimate on the arbitrarily chosen quadratic form, the enclosed volume of the estimate is chosen as the objective function to be maximized to obtain the optimal quadratic form. The result of the computations is a hyperellipsoidal estimate of the domain of attraction of $x = 0$ (i.e., the optimal quadratic estimate of the domain of attraction).

The equations describing the "paired-tracker" coarse pointing mode attitude control system of the OAO are derived and used as the test problem for determining the feasibility of the algorithm. This system is quite complex and highly nonlinear. It is described by nine state variables which are related through nonlinear differential equations representing the nonlinear mechanics, actuator saturation, and transcendental sensor relations. Two models are derived for the sensors and one is shown to introduce unnecessary complications in the stability analysis. It is also shown via simulation results that neither the linear approximation, nor the analytically attractive Popov approximation (linear part plus the saturation) adequately represents the system stability properties.

In developing the algorithm for application to this non-academic complex problem it was necessary to solve four major computational problems efficiently. These problems are: 1) generation of arbitrary positive definite matrices; 2) solution of the Liapunov matrix equation; 3) solution of the nine variable constrained minimum problem; and 4) solution of the forty-five variable maximization problem. The first was accomplished by developing a parameterization of the set of positive definite matrices. The second was solved by selecting from the four available techniques the one with the least error growth with increased

system dimension. The third was solved by devising an efficient random search tailored to the geometry of the problem. This was done subsequent to determining that the widely used gradient search techniques and penalty function methods are totally inapplicable to this problem. The fourth problem was partially solved via an "accelerated random search." Again, gradient techniques are totally inapplicable to this complex high dimensional problem.

The algorithm is shown to be feasible for solution of this complex nonlinear stability analysis problem. However, the state of the art in search techniques for complex high dimensional problems severely limits its immediate application as an analytical or operational tool. It is shown that compared to simulation, the only other currently available tool, it promises to produce a lower cost solution with specified confidence or conversely to produce a higher confidence result for a given cost.

In the course of examining the use of Luré-Liapunov functions to obtain an improved estimate it is shown that the Popov approximation to the system model cannot be completely analyzed with the available frequency domain techniques. It is also noted that computational aids are required if frequency domain techniques are to be effectively applied to a system of this dimension.

2. INTRODUCTION

This report describes the development of a numerical algorithm for estimating the domain of stability of complex nonlinear systems, and its application to a particular satellite attitude control problem. Specifically, the algorithm estimates the set of initial states from which a given system will settle on a desired equilibrium condition. That is, it estimates the domain of attraction of an equilibrium solution of the system of differential equations used to describe the physical system. The development was initiated because simulation showed that standard linear analysis failed to predict accurately the stability of the present NASA Orbiting Astronomical Observatory (OAO) coarse pointing mode control system. In addition, the algorithm promised results with higher confidence levels than reasonably possible via simulation. Simulation only provides representative operating records for a selected sample of the possible initial states and system parameters. The sample is necessarily limited by time and budget constraints. Thus, for a complex nonlinear system, the confidence in the conclusions drawn from simulation experiments is often less than desired.

The particular system design examined is the "paired-tracker" design of Doolin and Showman [1, 2]. This system is a very complex and highly nonlinear system that provides an excellent example for testing the true mettle of the algorithm. The model used to represent the system has nine state variables and accounts for the nonlinear characteristics of the actuators, vehicle, sensors, and error processor. The algorithm is based on the use of quadratic form Liapunov functions to estimate the domain of stability of the system in state space and to determine the quadratic form that maximizes the volume of the estimate. (This concept was first described in [3].) The result is a hyperellipsoid that is the optimal quadratic estimate of the domain of stability. The algorithm itself is applicable to a wide variety of complex nonlinear systems and is in no way limited to the attitude control system described here.

The objectives of the study were: 1) to determine the feasibility of using this method of stability analysis on a complex physical problem, and 2) to develop a new tool to be used in the design/analysis and operation of the specific system. The first objective arose from the conviction that it is utterly naive to assume, as is often done, that if a technique is shown to solve a few simple academic problems it can then easily be extended to

solve complex physical problems successfully. The second objective developed because it became apparent upon reviewing the design/analysis of the present OAO coarse pointing mode system that there were no applicable analytical techniques for stability analysis and that simulation could not yield the required level of confidence in the results. The nature of the system is such that the effect on system stability of new vehicle commands must be assessed on the ground during a flight and so the tool used in analysis/design would also likely be used in operations. Thus if the algorithm proved to be feasible and provided, as it should, more confidence than simulation results, it would become a significant tool in the OAO program.

The algorithm is based on Liapunov stability theory, the only available sufficiently general approach to nonlinear system stability analysis. The primary difficulty in application of the theory is the construction of an appropriate Liapunov function. This difficulty is eliminated by restricting consideration to positive definite quadratic form Liapunov functions. This restriction is substantial, but it results in an estimate that is always an ellipsoid in n-space, and is often a better estimate than those obtained with more complex functions [4, 5]. The fact that the estimate is always a hyperellipsoid means it is easier to visualize and interpret than other estimates, and some of the computations are simplified.

The report is organized as follows. In Section 3 the problem of estimating the domain of attraction of an equilibrium solution is formulated and the requisite parts of Liapunov stability theory are presented. The optimal quadratic estimate is formulated as a min-max problem and the structure of an algorithm to solve the problem is outlined. The solutions of four specific computational problems are presented, namely, the generation of positive definite matrices, solution of the Liapunov equation, solution of the minimum problem, and solution of the maximum problem.

Section 4 displays the formulation of the system state equations, a reduced state approximation, and an approximation suitable for Popov type analysis. The result of comparing these models by simulation is that neither the linear approximation nor the Popov approximation adequately represents the stability properties of the system.

The computational procedures and over-all program description are presented in Section 5. It is shown here that gradient search techniques are totally inapplicable to the min-max problem. The

method of interpreting the program results is described and some representative results are given for the complete nine dimensional model and the six dimensional reduced state model. The algorithm is then compared to simulation on the basis of cost to achieve a similar level of confidence.

In Section 6, the conclusions of the feasibility evaluation of the algorithm are presented, along with some conclusions on the state of the art in search techniques and in the stability analysis of complex nonlinear physical problems. This leads to identification of some problems requiring further attention.

The appendices present details of various parts of the study that were too complicated for the body of the report. Appendix A gives details of the system model derivation and the derivation of its approximations. Simulation results are illustrated in Appendix B. In Appendix C the linear model is analyzed and the Popov approximation is only partially analyzed because of the limitations in the current state of the art in frequency domain techniques. Appendix D reports details of solutions of computational problems and outlines reasons for inapplicability of gradient searches and penalty function techniques. The way to interpret the numerical results is described in Appendix E. In Appendix F, a variant on the algorithm is presented, along with an outline of the use of Luré-Liapunov functions for obtaining an improved estimate. Finally, Appendix G gives flow charts for the algorithm. FORTRAN IV program listings are available upon request from the Research Department, Grumman Aerospace Corporation, Bethpage, New York 11714.

3. OPTIMAL QUADRATIC ESTIMATION OF THE DOMAIN OF ATTRACTION

This section is devoted to describing the theory of optimal quadratic estimation of the domain of attraction, the structure of an algorithm for obtaining the estimate, and presenting solutions of associated computational problems. The reader is assumed to be familiar with fundamental Liapunov stability theory, say [6].

It is assumed that the physical system is described by a quasilinear vector differential equation of the form

$$\dot{x} = Ax + g(x) \tag{3-1}$$

where the dot denotes differentiation with respect to time t, $x = x(t)$ is the $n \times 1$ real state vector, A is the matrix of the linear part which is stable, $x(t) \equiv 0$ is an equilibrium solution, and $g(x)$ contains no linear terms, i.e., denoting the eigenvalues of A by $\lambda_i(A)$,

$$\begin{aligned} &\mathrm{Re}\left(\lambda_i(A)\right) < 0 \qquad (i = 1, 2, \ldots, n) \\ &g(0) = 0 \\ &\lim_{x \to 0} \frac{\|g(x)\|}{\|x\|} = 0 \ . \end{aligned} \tag{3-2}$$

These assumptions are not restrictive with respect to engineered systems. Since virtually all design techniques are based upon linear systems analysis, an engineered lumped parameter system will almost always be representable by (3-1) and meet the assumptions of (3-2). The latter simply state that the linearization of the system has its poles in the left half plane, the equilibrium solution ($\dot{x}_e \equiv 0$) can be shifted to the origin by the translation $x' = x - x_e$, and the power series expansion of $g(x)$ has no linear terms.

The domain of attraction $\mathcal{D}(0)$ of the equilibrium solution $x = 0$ is the set of initial states from which all trajectories settle to zero as time tends to infinity, i.e.,

$$\mathcal{D}(0) = \left\{x_o \,|\, x(t;\, x_o) \to 0 \quad \text{as} \quad t \to \infty\right\} \tag{3-3}$$

where $x(t,x_o)$ denotes the unique solution of (3-1) such that $x(0) = x_o$. The assumptions that A is a stable matrix and $g(x)$ has no linear part mean that $x = 0$ is an asymptotically stable solution of (3-1) and thus $\mathcal{D}(0)$ is a nonempty set. In the event that $\mathcal{D}(0)$ is the whole space, $x = 0$ is said to be globally asymptotically stable and there can be no other equilibrium solution. Many nonlinear systems are not globally asymptotically stable and the problem then is to determine $\mathcal{D}(0)$ for given system parameter values. This is the problem treated here.

The two most frequently used approaches to determining $\mathcal{D}(0)$ are the Zubov method and the LaSalle theory. The Zubov method requires solution of a partial differential equation which contains an arbitrary function. The solution is directly dependent on the arbitrary function and is usually obtained via a power series. If the solution can be obtained in closed form the domain $\mathcal{D}(0)$ is obtained exactly; however, this is rarely the case and each truncation of the series solution provides an estimate of $\mathcal{D}(0)$. The convergence of the series usually is nonuniform and is dependent on an arbitrary function. Often the first term of the series (which is a quadratic form) provides a better estimate than higher order estimates. Further, higher order estimates are hard to visualize and interpret. Finally, the series solution requires that $g(x)$ be expressed as a power series [4], which can be extremely tedious in a complex problem. This led to the concept [3, 5] of developing the optimal quadratic estimate which would provide suitable engineering estimates, easier visualization and interpretation, and simpler computation.

The LaSalle theory is summarized in his theorem on the extent of asymptotic stability [6].

> Theorem: Let $V(x)$ be a scalar function with continuous first partial derivatives. Let $\Omega_\ell = \{x \,|\, V(x) < \ell\}$ designate the region where $V(x) < \ell$. Assume that Ω_ℓ is bounded and that within Ω_ℓ:
>
> $$V(x) > 0 \quad \text{for} \quad x \neq 0$$
>
> $$\dot{V}(x) < 0 \quad \text{for} \quad x \neq 0 \,. \tag{3-4}$$

Then the origin is asymptotically stable, and above all, every solution in Ω_ℓ tends to the origin as $t \to \infty$.

Since the function $V(x)$ is at our disposal we choose the simplest one, i.e., a positive definite quadratic form,

$$V(x) = x^T P x \ , \qquad P > 0 \ . \tag{3-5}$$

Others [7, 8] have tried the Luré-Liapunov function

$$V(x) = x^T P x + \int_0^x g^T(x)\,dx \tag{3-6}$$

but use of (3-6) requires either proving positivity of (3-6) for given $g(x)$ or limiting $g(x)$ to the class of functions for which $V(x)$, in (3-6), is positive definite. Neither seems a suitable alternative. In any event, the estimate Ω_ℓ will be much more complex to visualize and interpret if (3-6) is used. The use of (3-5) guarantees that $V(x) > 0$ for $x \neq 0$, and that Ω_ℓ is bounded since Ω_ℓ is always a hyperellipsoid. Using (3-5) and (3-1) yields

$$\dot{V}(x) = - x^T Q x + 2 x^T P g(x) \tag{3-7}$$

where

$$A^T P + P A = - Q \ . \tag{3-8}$$

Equation (3-8) is called the Liapunov matrix equation. If Q is positive definite then $\dot{V}(x) < 0$ in a neighborhood of $x = 0$ since $g(x)$ is of the order of x^2. Given a positive definite Q and a stable A the solution P of (3-8) is always positive definite. Thus, it remains to specify ℓ such that $\dot{V}(x) < 0$ in Ω_ℓ. That is, find the largest ℓ such that the condition holds or, equivalently, since the ellipsoids Ω_ℓ are concentric and nested, find the least value of $V(x)$ on $\dot{V}(x) = 0$. The equivalence is indicated in Fig. 3-1.

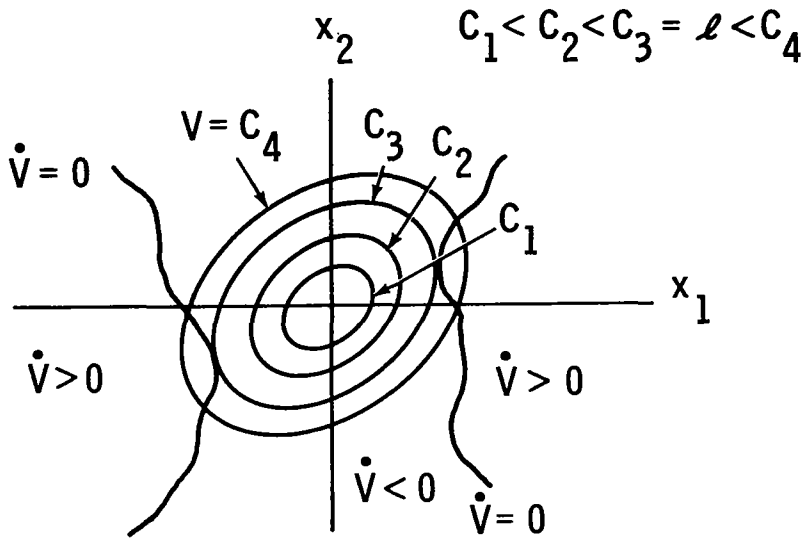


Fig. 3-1 Two Dimensional Representation of Problem Geometry

Thus, we define

$$\ell = \min_{x \in E} V(x)$$
$$E = \left\{ x \mid \dot{V}(x) = 0 \ , \ x \neq 0 \right\} \tag{3-9}$$

i.e., as the solution of a constrained minimum problem. The origin is excluded because it yields the trivial solution $\ell = 0$.

The value of ℓ and thus the estimate Ω_ℓ are functions of Q, through P and the Liapunov equation. To remove this dependence on an arbitrary matrix we define the optimal matrix, Q^o, to be the one which produces the estimate with the largest volume. The volume of the estimate is proportional to

$$J(Q) = \prod_{i=1}^{n} \left(\frac{\ell}{\lambda_i(P)} \right)^{\frac{1}{2}} = \left(\frac{\ell^n}{\det P} \right)^{\frac{1}{2}} \tag{3-10}$$

i.e., the product of the semiaxes of Ω_ℓ, and thus

$$J(Q^o) = \max_{Q > 0} J(Q) = \max_{Q > 0} \left(\frac{\ell^n}{\det P}\right)^{\frac{1}{2}} . \qquad (3\text{-}11)$$

The optimal quadratic estimate, Ω^o_ℓ, is obtained by solving (3-9) and (3-11), which form a min-max problem. The algorithm that solves this problem must, in the process, solve the following problems: 1) generate arbitrary elements, Q, of the set of positive definite matrices; 2) solve the Liapunov equation (3-8); 3) solve the minimum problem (3-9) fcr each Q; and 4) determine the optimal Q according to (3-11). The way in which ℓ and Q are related means that problem 4) must be solved by repeatedly solving problem 3). (Another approach to this problem based on selecting P matrices is given in Appendix F.)

The matrices Q are generated as positive definite matrices by recognizing their orthogonal similarity to a diagonal matrix with positive eigenvalues, i.e.,

$$Q = S^T \Lambda S \quad , \quad S^T S = I_n$$

$$\Lambda = \text{diag} \left\{\lambda_1, \ldots, \lambda_n\right\} , \ \lambda_i > 0 , \qquad i = 1, 2, \ldots, n . \qquad (3\text{-}12)$$

The matrices S are generated as a product of simple rotation matrices by utilizing a parameterization of unitary matrices given by Murnaghan [9]. (The details are given in Appendix D.) The $n \times n$ matrix Q is then specified by $n(n + 1)/2$ parameters (the number of free elements in a symmetric matrix) and formed by $\left(n(n - 1)/2\right) + 1$ matrix multiplications.

There are at present writing four methods of solving the Liapunov equation. Comparison of the four, based on increase in error and computation time with increase in dimension, happens to lead to selection of the least elegant approach. Although it is the most time consuming it suffers least from increase in error with increase in dimension. The solution arises from recognizing that if the Liapunov equation is written as a vector equation (P and Q reordered as $n^2 \times 1$ matrices) a simple pattern appears, viz.,

$$A_{mod} \begin{bmatrix} p_{11} \\ \vdots \\ p_{1n} \\ p_{21} \\ \vdots \\ p_{n1} \\ \vdots \\ p_{nn} \end{bmatrix} = - \begin{bmatrix} q_{11} \\ \vdots \\ q_{1n} \\ q_{2n} \\ \vdots \\ q_{n1} \\ \vdots \\ q_{nn} \end{bmatrix} , \tag{3-13}$$

where

$$A_{mod} = A^T \otimes I_n + I_n \otimes A^T , \tag{3-14}$$

$\otimes$ is the Kronecker product and I_n is the $n \times n$ identity matrix, i.e., if $n = 2$

$$A_{mod} = \left[\begin{array}{c|c} a_{11}I_2 + A^T & a_{21}I_2 \\ \hline a_{12}I_2 & a_{22}I_2 + A^T \end{array}\right] . \tag{3-15}$$

Thus, P is obtained by forming A_{mod}, calculating its inverse, calculating the elements of P according to (3-13) and restructuring P. (See Appendix D for details.)

The determination of an estimate, given the matrix Q, i.e., the calculation of ℓ via

$$\ell = \min_{x \in E} V(x)$$
$$E = \left\{ x \,|\, \dot{V}(x) = 0 \ , \ x \neq 0 \right\} \tag{3-9}$$

can be carried out via a gradient search-penalty function technique or a specially developed random search. In the first approach the constraints of the problem are eliminated by replacing them with penalty terms, i.e., by redefining ℓ as

$$\ell = \min_{x} \left[\left(V(x) \right)^{n_1} + k_1 \left(\dot{V}(x) \right)^2 + k_2 \left(\rho(x) \right)^2 \right] \qquad (3\text{-}16)$$

where $n_1 = 1$ or 2, k_1 and k_2 are positive, the second term is the penalty for straying from $\dot{V}(x) = 0$, and the third term is the penalty for nearing $x = 0$. In theory, if k_1 and k_2 tend to infinity then ℓ of (3-16) approaches ℓ of (3-9). In practice the selection of k_1 and k_2 is a very delicate matter since if they are too large the function $V(x)$ is "masked" and if too small the penalties are not severe enough and a search will wander away from the constraint surface and possibly toward the point $x = 0$. This is an important practical problem because as a rule the range of the functions in (3-16) is not known for a given domain of their arguments, even to orders of magnitude, without expending substantial additional computational effort. In any event, the term $\rho(x)$ tends to introduce unknown local minima near the origin, which, as any other local minima, act as a trap for a gradient type search. [One can visualize this by considering $\rho(x)$ to be an inverted cup in the bowl $V(x)$.] Gradient searches are inapplicable to this problem because of the many local minima, the superior attractiveness of the trivial solution at $x = 0$, and the difficulty of obtaining the gradient of the expression in parentheses in (3-16) either analytically or numerically.

The random search, which was initially developed to provide a method for certifying that the computed value was in fact the solution, actually became a more effective tool for solving the problem. It utilizes the basic geometry of the problem as portrayed in Fig. 3-1. A large value V_o is chosen, the ellipsoid $V(x) = V_o$ is constructed and points x are selected at random from the circumscribed box, then the logical pattern of Fig. 3-2 is followed.

The net result is that the search is always conducted in a succeedingly smaller box circumscribed about an ellipsoid and the random search looks for a point where $\dot{V}(x) > 0$ from which a one dimensional search along the line from x to the origin proceeds to find $\dot{V}(x) = 0$. At that point a new smaller ellipsoid is defined and the procedure is repeated.

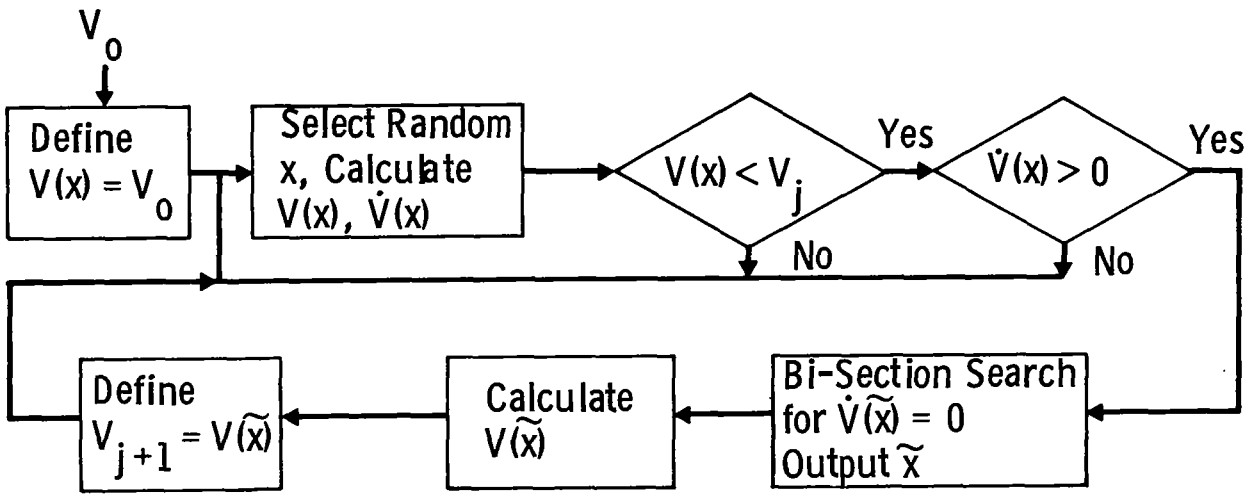


Fig. 3-2 Schematic of Random Search Procedure

The determination of the optimal matrix, Q^o, is much more difficult because of the higher dimension of the problem, $n(n + 1)/2$ versus n, and the lack of any knowledge of the problem geometry. This problem was attacked via a modification of the "accelerated random search" described by Barron [10]. Basically, the process is a one dimensional deterministic search along a randomly selected direction. The procedure is to select, from a specified distribution, a set of perturbations on a starting point; if an improvement is achieved, continue searching in the same direction and double the step size each time until no further improvement is obtained. If no improvement is obtained on the first step, reverse direction and proceed as above. If neither direction yields an improvement, select a new set of perturbations. The acceleration is obtained from doubling the step sizes in the successful direction. In addition, the distribution is "narrowed down" as success is achieved; this provides a finer search toward the end. We have added the capability to "widen" the distribution if no success is achieved after a certain number of trials. This makes possible "jumping out" of local minima in which the search

may become trapped if the perturbations are too small. The details of these searches may be found in Appendix D. Gradient searches are inapplicable to the maximization problem because of its dimension. In the 45 dimensional space of Q, 45 random samples produce more useful global information than the 45 perturbations required to numerically evaluate the gradient of (3-10)at one point, which cannot be obtained analytically.

4. MODELING OF THE OAO "PAIRED-TRACKER" COARSE POINTING MODE ATTITUDE CONTROL SYSTEM

This section presents a summary of the derivation of the system model, the formulation of the state equation in the required form, and some approximations to the state equation.

A model is needed to represent the actual hardware and spacecraft dynamics in mathematical form. Although modeling of the OAO had been done prior to this study, a rederivation was performed to provide a physical "feel" for the model and an appreciation of the approximations necessary to create a useful and usable model.

A digital simulation is used to verify stability or lack of it for various cases for comparison with the estimate produced by the algorithm. It is also used to compare the various simplified models with the principal one.

The derivation of the system model will be considered first. It will be seen that the choice of sensor model greatly affects the difficulty in performing the analysis. The basic block diagram is given in Fig. 4-1.

The OAO coarse pointing mode attitude control system uses inertia wheels to supply control torques and momentum storage. Vehicle attitude is sensed by star trackers, whose output is processed and put through a compensator to generate attitude plus derived rate information. The compensator output drives the momentum wheel motors. In effect this system is a momentum regulator with an equilibrium described by zero body rates and an attitude at or near the one desired.

To derive a model that is of practical use some basic assumptions must be made. The assumptions that follow lead to simplifications in the model, but they must be justified by a demonstration that the model behaves essentially like the real system.

1. The vector between the spacecraft and a guide star expressed in inertial coordinates is assumed to be constant independent of orbit position. That is to say, parallax effects are neglected. This assumption is borne out by the fact that the parallax to the nearest star (not our sun) is only 0.75 sec of arc.

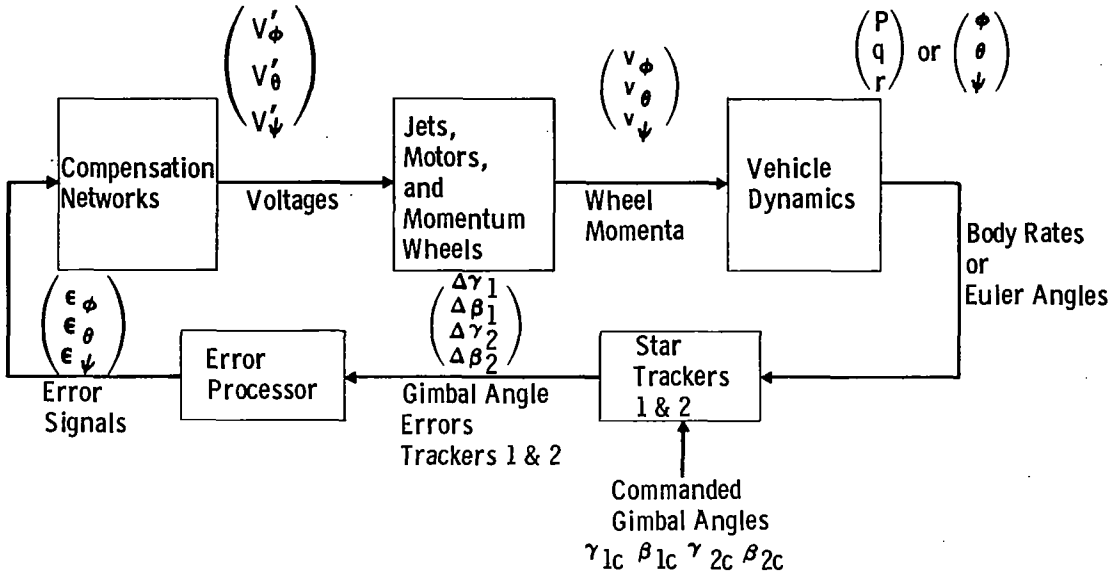


Fig. 4-1 Block Diagram of Basic Model

2. The dynamics of the gimbaled star tracker, basically a lag, is neglected. Analysis has shown that the lag, if included, has no effect on spacecraft dynamics. The lag break occurs approximately one decade beyond the region of spacecraft response.

3. The saturation and quantization in the star tracker readout (digitizer logic unit, DLU) are ignored. This assumption was an imposed ground rule of the study, and was based on the expectation of using a DLU with a larger linear range. The quantization has negligible effect on stability.

4. The gyroscopic torques of the control wheels are neglected. This assumption is verified by study and simulation of the OAO.

5. The inertia is assumed to be spherical; in particular, the torque coupling of rate and acceleration due to products of inertia is assumed to be zero. Once again simulation and the fact that $I_{ij}/I_{jj} < 0.01$ bear out this assumption, where I_{ij} are elements of the inertia tensor.

Prior to discussing each block of the figure it is necessary to define some coordinate systems. An inertial reference frame, fixed in space, is defined as X_r, Y_r, Z_r, where X_r is the line of sight to the target. A body frame, X_b, Y_b, Z_b, aligned with the control axes, is related to the inertial frame by a conventional Euler transformation described by ϕ, θ, ψ, the roll, pitch, and yaw Euler angles. Each tracker has its own reference frame, X_{TR}, Y_{TR}, Z_{TR}, which is fixed in the body and is related to the body frame by a simple transformation. The tracker axis frame, X_T, Y_T, Z_T, is then related to the tracker reference frame by the angles α, β, γ (rotations about the tracker optical, inner gimbal, and outer gimbal axes). Thus all axes of importance are related to a fixed inertial frame. Each individual block of the system will now be discussed. Appendix A presents a detailed description of the coordinate systems and the blocks considered below.

First consider the block of trackers. The two basic models and their relative merits are discussed at length in [11]. The angle model is derived from the fact that the line of sight to a star is fixed in inertial space. The rate model is derived by equating the vehicle rotational rate as expressed in the inertial frame and the tracker frame, which is by definition also an inertial frame. Basically, the angle model for tracker 1 is described by

$$\Delta\beta_1 = \sin^{-1}(c\psi\, c\theta\, s\beta_{1C} + s\psi\, c\theta\, c\gamma_{1C}\, c\beta_{1C} + s\theta\, s\gamma_{1C}\, c\beta_{1C}) - \beta_{1C} \tag{4-1}$$

$$\Delta\gamma_1 = \tan^{-1}\left(\frac{-(s\psi s\phi + c\psi s\theta c\phi)\, s\beta_{1C} + (c\psi s\phi - s\psi s\theta c\phi)\, c\gamma_{1C}\, c\beta_{1C} + c\theta c\phi s\gamma_{1C}\, c\beta_{1C}}{-(s\psi c\phi - c\psi s\theta s\phi)\, s\beta_{1C} + (c\psi c\phi + s\psi s\theta s\phi)\, c\gamma_{1C}\, c\beta_{1C} - c\theta s\phi s\gamma_{1C}\, c\beta_{1C}}\right) - \gamma_{1C}$$

where $\Delta\beta$, $\Delta\gamma$ are gimbal angle errors and ϕ, θ, ψ must be derived from p, q, r in an Euler integration block. [Note that $s(\cdot) \equiv \sin(\cdot)$, $c(\cdot) \equiv \cos(\cdot)$, and $t(\cdot) \equiv \tan(\cdot)$.] This block solves three simultaneous nonlinear differential equations in ϕ, θ, ψ with inputs p, q, r and is independent of the number of trackers in use.

The rate model for tracker 1 is

$$\dot{\Delta\beta}_1 = s(\Delta\gamma_1 + \gamma_{1c})q + c(\Delta\gamma_1 + \gamma_{1c})r$$

$$\dot{\Delta\gamma}_1 = p - t(\Delta\beta_1 + \beta_{1c})c(\Delta\gamma_1 + \gamma_{1c})q + t(\Delta\beta_1 + \beta_{1c})s(\Delta\gamma_1 + \gamma_{1c})r \qquad (4\text{-}2)$$

Attitude control of the spacecraft necessitates a minimum of two trackers. Thus this model at best requires solving four simultaneous differential equations in β_i, γ_i, β_j, γ_j $i \neq j = 1,2,3,4$, with p, q, r as inputs. It also requires a system order of at least one greater than the angle model. The rate model will also cause a column of zeros in the matrix of the linear part of the system state equation. This leads to difficulties because a critical matrix will not yield a positive definite solution P of the Liapunov equation for any positive definite Q.

The next block in the figure represents the error processor. The processor used here is the Doolin and Showman "partial processor," which is based on an attempt to produce an uncoupled error signal using only resolvers and analog summers. This processor is described by

$$\begin{bmatrix} \epsilon_\phi \\ \epsilon_\theta \\ \epsilon_\psi \end{bmatrix} = \begin{bmatrix} a_{11} & 1 & a_{13} \\ d_{12}c(\gamma_{2c} + \Delta\gamma_2) & 0 & d_{12}c(\gamma_{1c} + \Delta\gamma_1) \\ -d_{12}s(\gamma_{2c} + \Delta\gamma_2) & 0 & -d_{12}s(\gamma_{1c} + \Delta\gamma_1) \end{bmatrix} \begin{bmatrix} \Delta\beta_1 \\ \Delta\gamma_1 \\ \Delta\gamma_2 \end{bmatrix} \qquad (4\text{-}3)$$

The next block in the figure represents the compensator which is a lead-lag. The transfer function is

$$\frac{V_i'}{\epsilon_i} = G_c(s) = K_c \frac{(\tau_1 + \tau_2)s + 1}{\tau_2 s + 1} \qquad i = \phi, \theta, \psi \qquad (4\text{-}4)$$

which is modeled as shown in Fig. 4-2 in order to identify the state variables ω_i easily. The state equation is

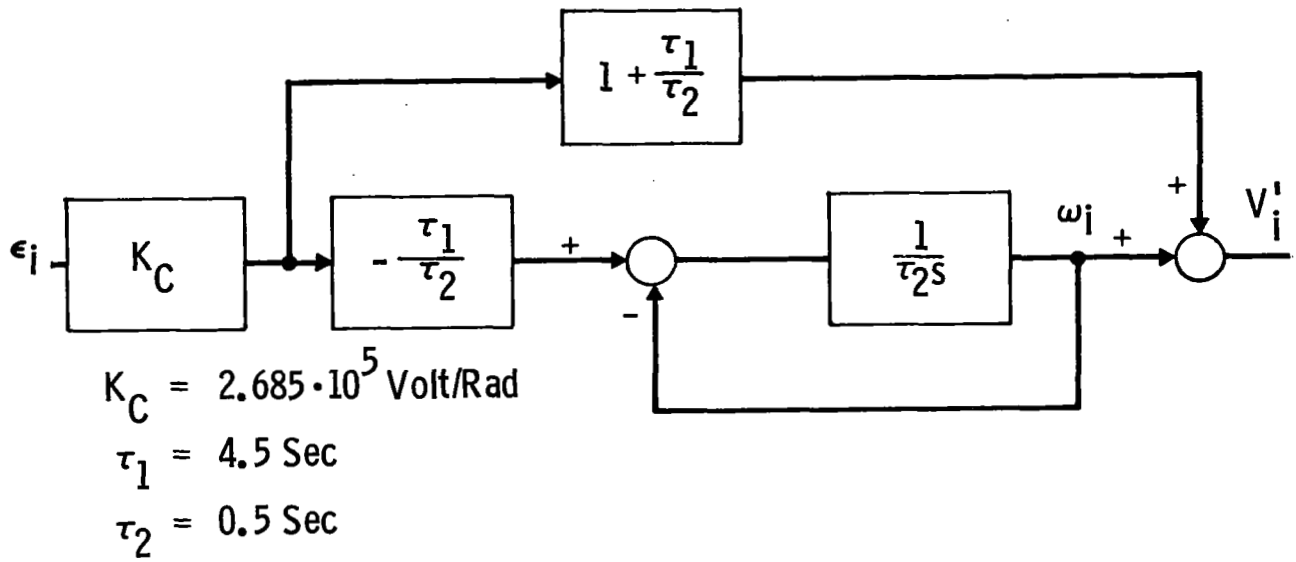


Fig. 4-2 Compensator Model Identifying Compensator State Variable

$$\dot{\omega}_i = -\frac{1}{\tau_2}\omega_i - \frac{K_c\tau_1}{\tau_2^2}\epsilon_i \qquad i = \phi, \theta, \psi \tag{4-5}$$

with output voltage

$$V_i' = \omega_i + K_c(1 + \tau_1/\tau_2)\epsilon_i \qquad i = \phi, \theta, \psi \tag{4-6}$$

The motor momentum wheel block comprises a saturation and ideal inertia wheel. The saturator is described in Fig. 4-3.

$$V_i'' = f(V_i') = 26\ \mathrm{sat}(V_i'/26) \tag{4-7}$$

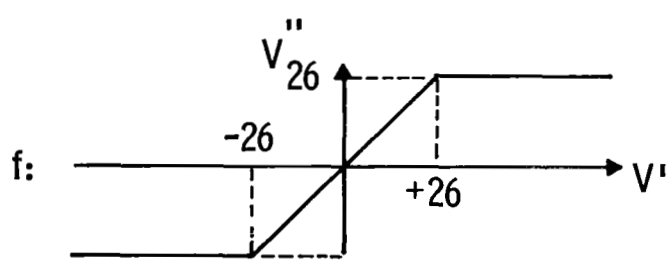


Fig. 4-3 Motor Saturator Function

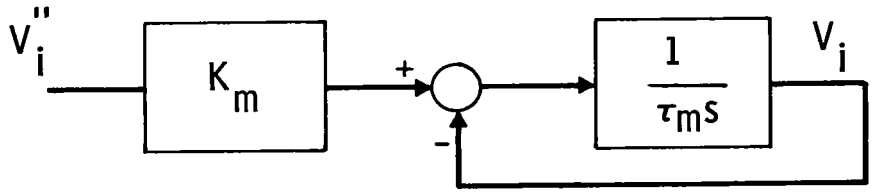
The wheel can be configured as a torquer

$$\frac{\hat{H}_{\omega_i}(s)}{V_i''(s)} = G_{MH}(s) = \frac{K_m s}{\tau_m s + 1}, \qquad i = \phi, \theta, \psi \tag{4-8}$$

or a momentum storage device

$$\frac{v_i(s)}{V_i''(s)} = G_M(s) = \frac{K_m}{\tau_m s + 1}, \qquad i = \phi, \theta; \psi . \tag{4-9}$$

Assumptions (4) and (5), which uncouple the vehicle equations allow them to be integrated so that the vehicle is described by a momentum balance and the wheel momentum v_i is the vehicle input. Thus, the momentum configuration is used and described by the block diagram of Fig. 4-4



$$K_m = \frac{1}{13} \ \frac{\text{Ft-Lb-Sec}}{\text{Volt}} \qquad \tau_m = 76.8 \text{ Sec}$$

Fig. 4-4 Motor and Momentum Wheel

yielding

$$\dot{v}_i = -\frac{1}{\tau_m} v_i + \frac{K_m}{\tau_m} V_i'', \qquad i = \phi, \theta, \psi \tag{4-10}$$

The vehicle is simply represented by a balance between wheel momentum v_i, corresponding to vehicle momentum components Ip, Iq, or Ir, and Ih_i^o the corresponding total momentum assumed to be constant.

$$p = -\frac{v_\phi}{I} + h_\phi^o$$

$$q = -\frac{v_\theta}{I} + h_\theta^o \qquad (4\text{-}11)$$

$$r = -\frac{v_\psi}{I} + h_\psi^o$$

The final block (required only for the tracker angle model) is the Euler block described by

$$\dot{\phi} = p + q\ t\theta\ s\phi + r\ t\theta\ c\phi$$

$$\dot{\theta} = q\ c\phi - r\ s\phi \qquad (4\text{-}12)$$

$$\dot{\psi} = q\ \frac{s\phi}{c\theta} + r\ \frac{c\phi}{c\theta}\ .$$

Figure 4-5 presents the combined model in block form with the angle model for star trackers 1 and 2.

The state equations below are a summary of the work above using the angle model and assumptions 1-5.

$$\dot{\phi} = \left(-\frac{v_\phi}{I} + h_\phi^o\right) + t\theta s\phi \left(-\frac{v_\theta}{I} + h_\theta^o\right) + t\theta c\phi \left(-\frac{v_\psi}{I} + h_\psi^o\right)$$

$$\dot{v}_\phi = -\frac{1}{\tau_m} v_\phi + \frac{K_m}{\tau_m} f\left(\omega_\phi + K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon_\phi\right) \qquad (4\text{-}13)$$

$$\dot{\omega}_\phi = -\frac{1}{\tau_2}\omega_\phi - \frac{K_c\tau_1}{\tau_2^2}\epsilon_\phi$$

$$\dot{\theta} = c\phi\left(-\frac{v_\theta}{I} + h_\theta^o\right) - s\phi\left(-\frac{v_\psi}{I} + h_\psi^o\right)$$

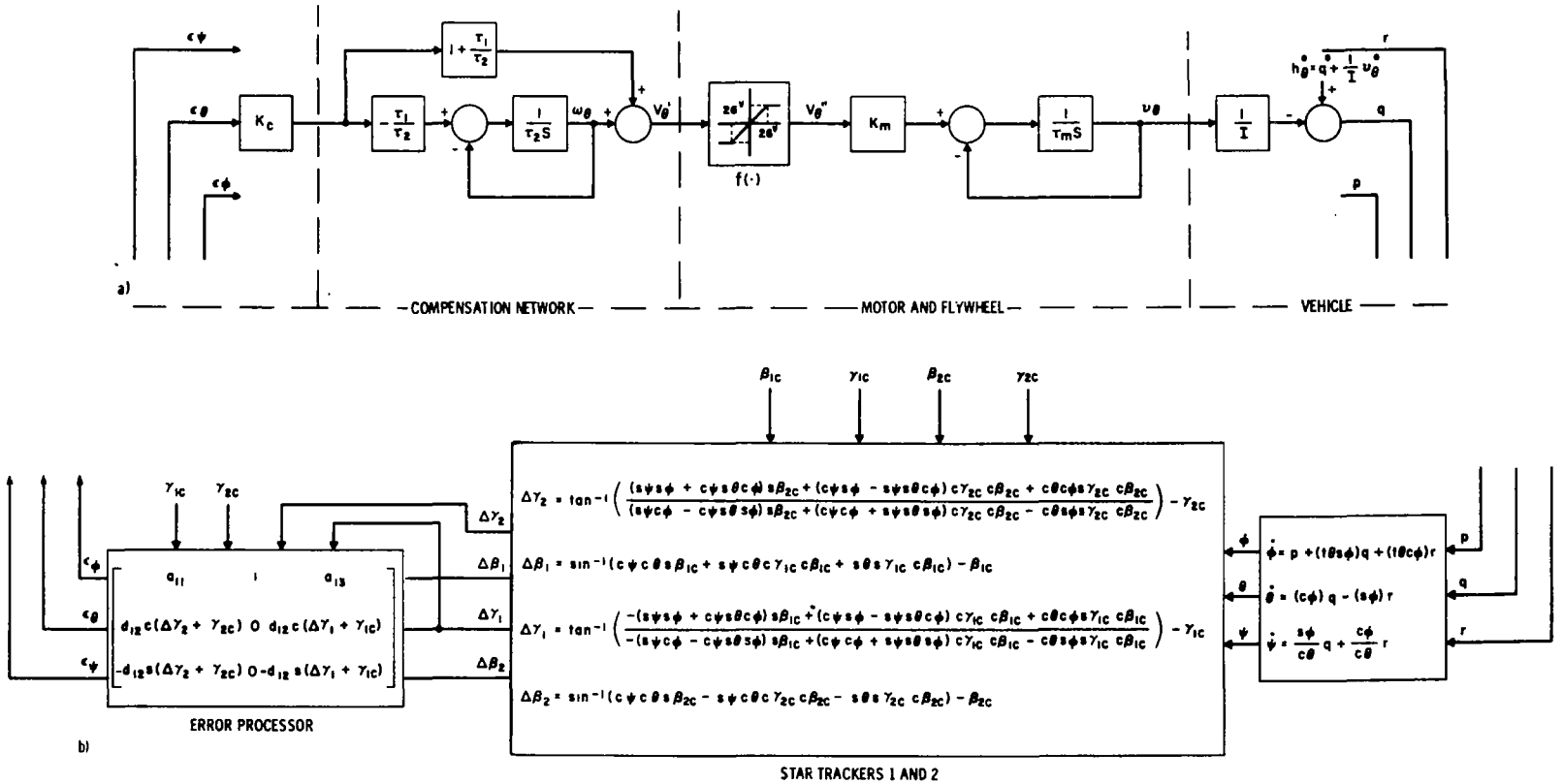


Fig. 4-5 Typical Forward Channel a) Without Wheel Gyroscopic Torques; and Feedback Path b) Based on Gimbal Angle Equations

$$\dot{v}_\theta = -\frac{1}{\tau_m} v_\theta + \frac{K_m}{\tau_m} f\left(\omega_\theta + K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon_\theta\right)$$

$$\dot{\omega}_\theta = -\frac{1}{\tau_2}\omega_\theta - \frac{K_c\tau_1}{\tau_2^2}\epsilon_\theta$$

$$\dot{\psi} = \frac{s\phi}{c\theta}\left(-\frac{v_\theta}{I} + h_\theta^o\right) + \frac{c\phi}{c\theta}\left(-\frac{v_\psi}{I} + h_\psi^o\right) \qquad (4\text{-}13)\ (\text{Cont.})$$

$$\dot{v}_\psi = -\frac{1}{\tau_m} v_\psi + \frac{K_m}{\tau_m} f\left(\omega_\psi + K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon_\psi\right)$$

$$\dot{\omega}_\psi = -\frac{1}{\tau_2}\omega_\psi - \frac{K_c\tau_1}{\tau_2^2}\epsilon_\psi$$

where the relations for ϵ_ϕ, ϵ_θ, ϵ_ψ, $\Delta\beta_1$, $\Delta\beta_2$, $\Delta\gamma_1$, and $\Delta\gamma_2$ are given in Fig. 4-5.

The state equation is obviously highly nonlinear and nine dimensional with the form

$$\dot{x} = F(x) \ . \qquad (4\text{-}14)$$

The analysis requires the state equation to be of the form

$$\dot{x}' = Ax' + g(x')$$

$$g(x_e') = 0 \qquad (4\text{-}15)$$

$$x_e' \equiv 0 \ .$$

To cast the equations in this form we define x' as

$$x' = x - x_e \qquad (4\text{-}16)$$

where x_e is the equilibrium solution, i.e., $F(x_e) \equiv 0$, and

$$A_{ij} = \frac{\partial F_i(x')}{\partial x'_j}\Big|_{x'=0}, \qquad i,j = 1, 2, \ldots, 9$$

$$g(x') = F(x') - Ax' \tag{4-17}$$

The elements of A are complicated functions of the system parameters and the equilibrium values of the attitude angle variables, ϕ_e, θ_e, ψ_e. These equilibria are given approximately as a linear function of the h_i^o where the coefficients are nonlinear functions of the commanded gimbal angles.

The A_{ij} for $\phi_e = \theta_e = \psi_e = 0$ are far simpler in form than the general case. Since ϕ_e, θ_e, ψ_e are quite small we believed that it might not be necessary to use the general form. To corroborate this, a program that calculates the eigenvalues of A for zero and peak h_i^o's over a range of command angles was written. The computational results indicated that, at worst, differences in the respective eigenvalues, both real and imaginary parts, occur in the fifth significant figure.

The matrix A is required for the solution of the Liapunov equation. In the course of solution it became evident that the large differences in magnitude of various elements of A caused numerical problems precluding an accurate solution. To combat this, the state was reformulated in a nondimensional form (x''), which yielded an A that permitted accurate solution of the Liapunov equation. Appendix A presents a detailed description of the model derivation.

The zero offset ($\phi_e = \theta_e = \psi_e = 0$) nondimensionalized state equation is presented in Fig. 4-6. It is this form that illustrates the linear uncoupling achieved by using the Ames "paired tracker" model. It should be noted that the complete state equations are not uncoupled. It is for this reason and because of the failure of linear analysis that a nonlinear analysis was performed. Simulation results, discussed below, substantiate the necessity of the nonlinear approach.

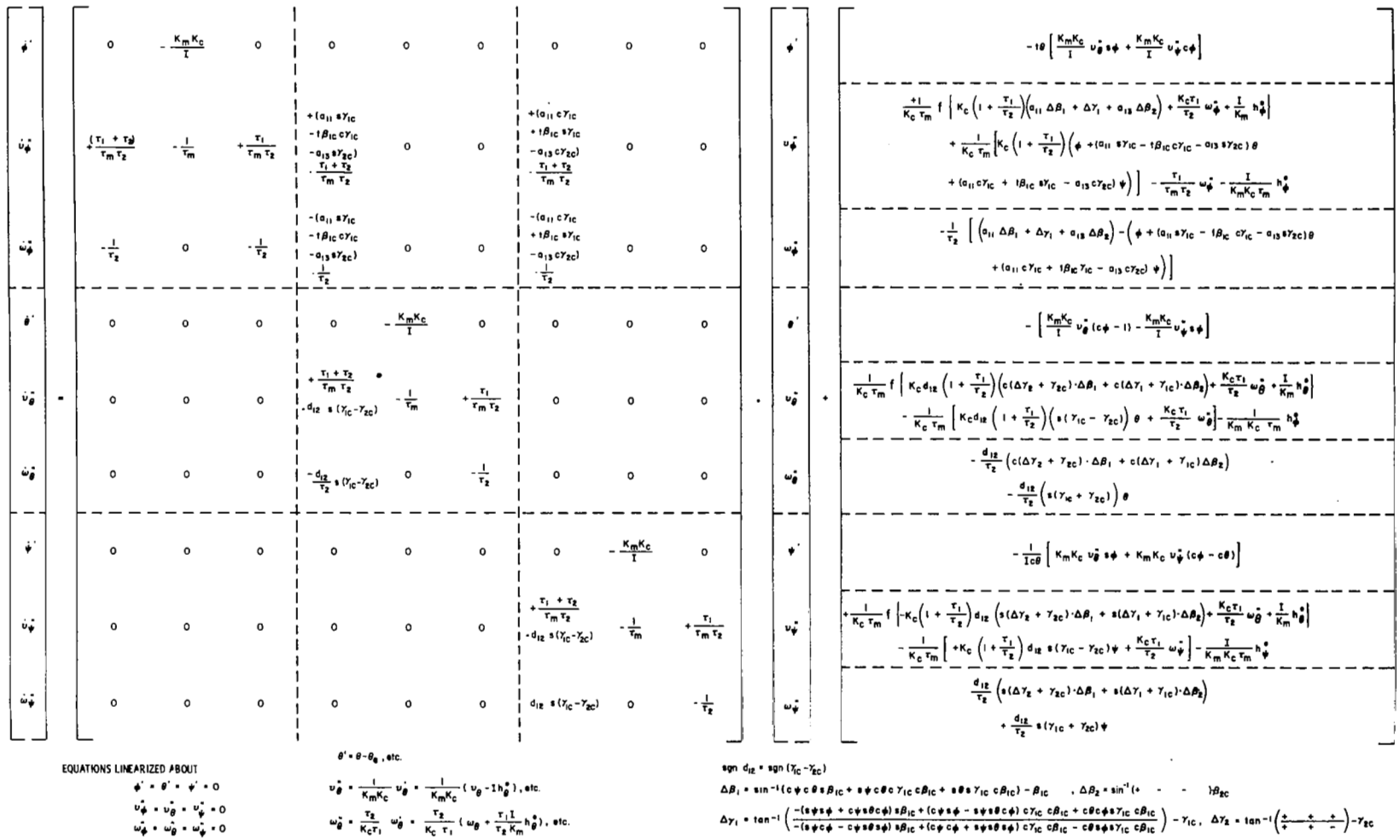


Fig. 4-6 Nondimensional State Equations Based on Tracker Angle Model — Offset Neglected

Three other models were also derived. Two of these, which were derived in the hope of simplifying the analysis, are approximations of the basic model. The third is a simple variation that tries to model a limiting of voltage in the compensator. Figures 4-8, 4-9, and 4-10, present the block diagram and state equations for each model. Blocks which are unchanged from the basic model (Fig. 4-5) are simply named. The basic model will for simplicity be labeled AN (all nonlinearities).

The first model studied ignores the compensator lag and thus the state variables ω_i, thereby reducing the dimensions of the state from nine to six, (labeled 6D). The lag break, like that of the tracker lag dynamics of assumption 2 , is basically beyond the spacecraft dynamic region. The approximate compensator transfer function becomes

$$G_c(s) \approx K_c(\tau_1 s + 1) \tag{4-18}$$

with model (Fig. 4-7)

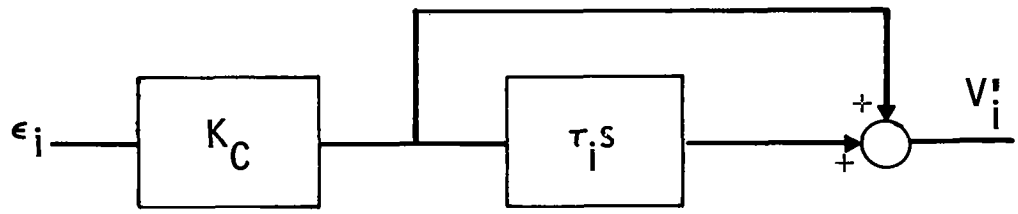
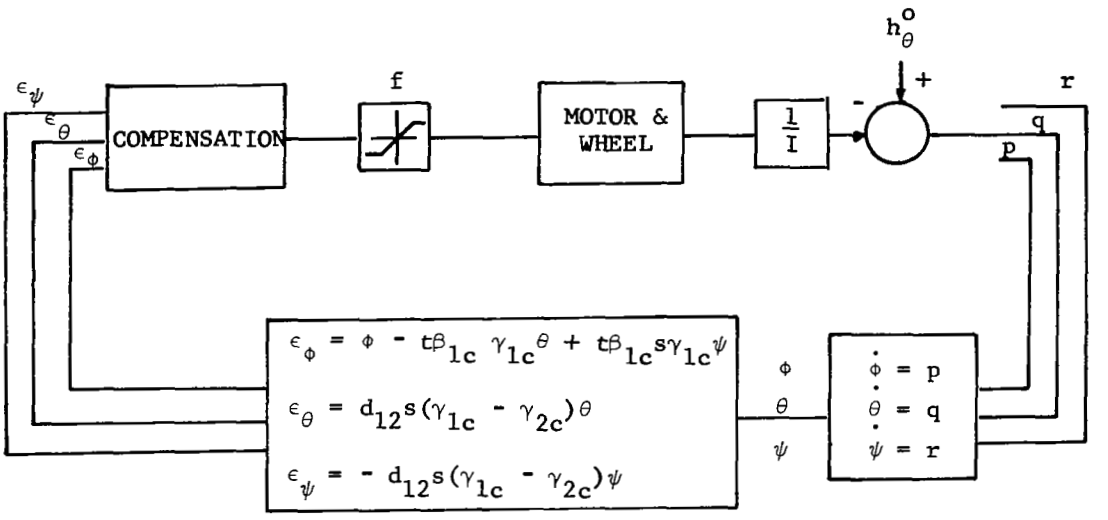


Fig. 4-7 Approximate Compensator Model

and equation

$$V_i' = K_c(\tau_1 \dot{\epsilon}_i + \epsilon_i) \quad , \quad i = \phi, \theta, \psi \tag{4-19}$$

A six dimensional state greatly reduced the dimensions of the space for J(Q) optimization, from 45 to 21. The benefits thus derived are discussed in Section 5.



$$\dot{\phi} = -\frac{v_\phi}{I} + h_\phi^o$$

$$\dot{v}_\phi = -\frac{v_\phi}{\tau_m} + \frac{K_m}{\tau_m} f\left(K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon_\theta + \omega_\phi\right)$$

$$\dot{\omega}_\phi = -\frac{\omega_\phi}{\tau_2} - \frac{K_c\tau_1}{\tau_2^2}\epsilon_\phi$$

$$\dot{\theta} = -\frac{v_\theta}{I} + h_\theta^o$$

$$\dot{v}_\theta = -\frac{v_\theta}{\tau_m} + \frac{K_m}{\tau_m} f\left(K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon_\theta + \omega_\theta\right)$$

$$\dot{\omega}_\theta = -\frac{\omega_\theta}{\tau_2} - \frac{K_c\tau_1}{\tau_2^2}\epsilon_\theta$$

$$\dot{\psi} = -\frac{v_\psi}{I} + h_\psi^o$$

$$\dot{v}_\psi = -\frac{v_\psi}{\tau_m} + \frac{K_m}{\tau_m} f\left(K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon_\psi + \omega_\psi\right)$$

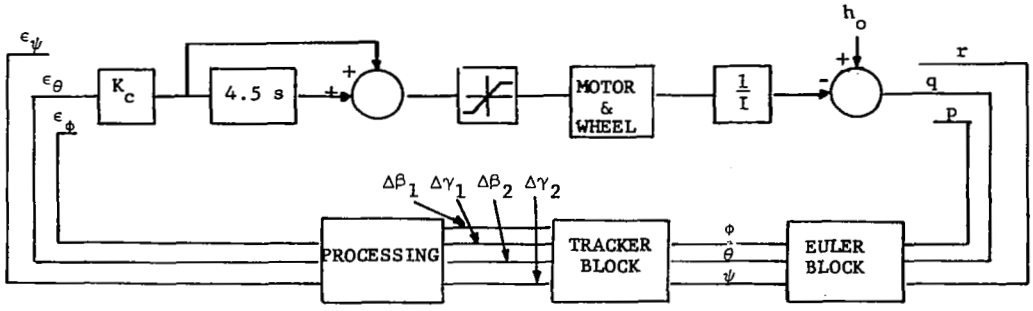
$$\dot{\omega}_\psi = -\frac{\omega_\psi}{\tau_2} - \frac{K_c\tau_1}{\tau_2^2}\epsilon_\psi$$

Note:

1) ϵ_ϕ, ϵ_θ, ϵ_ψ given in block above - linear functions of ϕ, θ, ψ

2) If block only titled it is identical to block in exact model - Fig. 4-5

Fig. 4-8 Motor Saturation Only Nonlinearity -- MV



$$\dot{\phi} = h_\phi^o - \frac{1}{I} v_\phi + (t\theta s\phi) \cdot \left(h_\theta^o - \frac{1}{I} v_\theta\right) + (t\theta c\phi) \cdot \left(h_\psi^o - \frac{1}{I} v_\psi\right)$$

$$\dot{v}_\phi = -\frac{1}{\tau_m} v_\phi + \frac{K_m}{\tau_m} f\left(K_c(\epsilon_\phi + 4.5\ \dot{\epsilon}_\phi)\right)$$

$$\dot{\theta} = c\phi \cdot \left(h_\theta^o - \frac{1}{I} v_\theta\right) - s\phi \cdot \left(h_\psi^o - \frac{1}{I} v_\psi\right)$$

$$\dot{v}_\theta = -\frac{1}{\tau_m} v_\theta + \frac{K_m}{\tau_m} f\left(K_c(\epsilon_\theta + 4.5\ \dot{\epsilon}_\theta)\right)$$

$$\dot{\psi} = \left(\frac{s\phi}{c\theta}\right)\left(h_\theta^o - \frac{1}{I} v_\theta\right) + \left(\frac{c\phi}{c\theta}\right)\left(h_\psi^o - \frac{1}{I} v_\psi\right)$$

$$\dot{v}_\psi = -\frac{1}{\tau_m} v_\psi + \frac{K_m}{\tau_m} f\left(K_c(\epsilon_\psi + 4.5\ \dot{\epsilon}_\psi)\right)$$

$$\dot{\epsilon}_\phi = PJ_4$$

$$\dot{\epsilon}_\theta = d_{12}\Big[c(\Delta\gamma_2 + \gamma_{2c}) \cdot PJ_1 + c(\Delta\gamma_1 + \gamma_{1c}) \cdot PJ_2 - s(\Delta\gamma_2 + \gamma_{2c}) \cdot \Delta\beta_1 \cdot PJ_3 - s(\Delta\gamma_1 + \gamma_{1c}) \cdot \Delta\beta_2 \cdot PJ_4\Big]$$

$$\dot{\epsilon}_\psi = -d_{12}\Big[s(\Delta\gamma_2 + \gamma_{2c}) \cdot PJ_1 + s(\Delta\gamma_1 + \gamma_{1c}) \cdot PJ_2 + c(\Delta\gamma_2 + \gamma_{2c}) \cdot \Delta\beta_1 \cdot PJ_3 + c(\Delta\gamma_1 + \gamma_{1c}) \cdot \Delta\beta_2 \cdot PJ_4\Big]$$

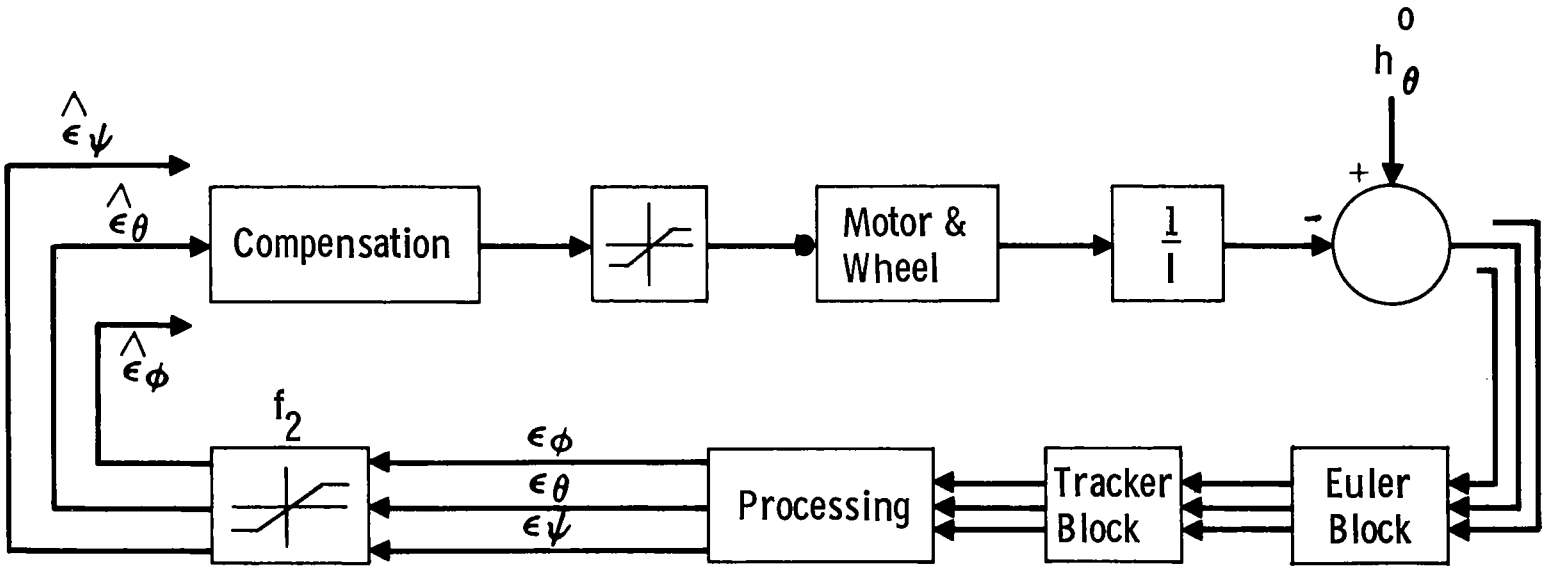
$$PJ_1 \equiv s(\Delta\gamma_1 + \gamma_{1c}) \cdot \left(h_\theta^o - \frac{v_\theta}{I}\right) + c(\Delta\gamma_1 + \gamma_{1c}) \cdot \left(h_\psi^o - \frac{v_\psi}{I}\right)$$

$$PJ_2 \equiv -s(\Delta\gamma_2 + \gamma_{2c}) \cdot \left(h_\theta^o - \frac{v_\theta}{I}\right) - c(\Delta\gamma_2 + \gamma_{2c}) \cdot \left(h_\psi^o - \frac{v_\psi}{I}\right)$$

$$PJ_3 \equiv \left(h_\phi^o - \frac{v_\phi}{I}\right) + t(\Delta\beta_2 + \beta_{2c}) \cdot c(\Delta\gamma_2 + \gamma_{2c}) \cdot \left(h_\theta - \frac{v_\theta}{I}\right) - t(\Delta\beta_2 + \beta_{2c}) \cdot s(\Delta\gamma_2 + \gamma_{2c}) \cdot \left(h_\psi^o - \frac{v_\psi}{I}\right)$$

$$PJ_4 \equiv \left(h_\phi^o - \frac{v_\phi}{I}\right) - t(\Delta\beta_1 + \beta_{1c}) \cdot c(\Delta\gamma_1 + \gamma_{1c}) \cdot \left(h_\theta^o - \frac{v_\theta}{I}\right) + t(\Delta\beta_1 + \beta_{1c}) \cdot s(\Delta\gamma_1 + \gamma_{1c}) \cdot \left(h_\psi^o - \frac{v_\psi}{I}\right)$$

Fig. 4-9 Compensator Lag Neglected Model -6D



The State Equations are Identical to Those of the "Exact Model" (Eq 4-13) Except ϵ_i $(i = \phi, \theta, \psi)$ is Replaced by $\hat{\epsilon}_i$ $(i = \phi, \theta, \psi)$ Where

$$\hat{\epsilon}_i = K \text{ Sat } (\epsilon_i/K) \qquad i = \phi, \theta, \psi$$

$$K = 100 \text{ Volts}$$

Fig. 4-10 Exact Model With Error Signal Limiting - ANL

The second model eliminates all nonlinearities except motor saturation and is labeled MV. Thus it is in all respects but one (the saturations) the same as the Doolin and Showman model of [2]. Moreover, it is identical in regard to channel uncoupling, which is the essence of their model. The motor voltage saturation is preserved because its small linear region seems to indicate that it is the dominant nonlinearity.

Finally, there is a variation of the basic model which simply places a saturator after $(\epsilon_\phi, \epsilon_\theta, \epsilon_\psi)$ to reduce the high voltage input to the compensator; it is labeled ANL. This model causes a large reduction in effective damping (rate lead) and closely approximates performance with a DLU. This subject, however, is not considered in the present stability analysis.

The simulation results are discussed in detail in Appendix B. Given a tracker case and initial conditions for which the basic model has an asymptotically stable equilibrium, there is a general similarity of trajectories of the various models. That is to say, the dynamics are basically the same.

There exist, however, tracker cases for which the basic model, the 6-D, and basic model with error limiting (which are identical in the feedback path) are unstable, but for which the motor saturation only model is stable. This phenomenon is discussed at length in Appendix B. The important fact is that this definitely implies that a model based upon linearizing the feedback path is not valid for stability analysis. In fact it is coupling in nonlinear feedback, as opposed to its nonlinear form, that may cause grave differences in performance between this approximation and the full nonlinear model.

5. COMPUTATION — PROGRAMS, PROBLEMS, RESULTS

Formidable computational problems are encountered in implementing the techniques of Section 3. Initially, four individual tasks were foreseen: 1) generate $n \times n$ positive definite symmetric matrices Q; 2) solve the Liapunov equation $A^TP + PA = -Q$ for P; 3) search the state space for the minimum of $V(x)$ on $\dot{V}(x) = 0$, $x \neq 0$; and 4) search the Q space for the Q^o that maximizes $J(Q)$. In the initial formulation, it was assumed that the problem of searching the space of parameters generating Q would be solved by the search technique of task 3.

Task 1, the generation of a positive definite Q matrix, was achieved, as summarized in Appendix D, part (ii). Generation of each 9×9 positive definite matrix Q requires specification of 45 parameters which characterize the matrix, and 37 matrix multiplications to form Q.

Various algorithms for the achievement of task 2, the solution of the Liapunov equation (3-8), have been presented in the literature. A comprehensive evaluation of four of these techniques appears in [12] and in Appendix D, part (i) of this report. One conclusion of [12] is that the method used here suffers least in accuracy deterioration with dimension increase while it suffers most with respect to increase of computing time with dimension. Methods of solving the Liapunov equation are at present well understood so that task 2 was completed in an entirely satisfactory manner.

The third task was by far the most challenging and required the most effort. Development of the search technique proceeded from first trying a gradient search, which is described in Appendix D, part (iii), to finally developing the random search, which is outlined in Appendix D, part (iv). Despite all claims made by various workers who have investigated gradient search procedures, all of our efforts in this area that utilized penalty functions of various sorts were ineffective. The gradient procedure reliably found the trivial global solution of (3-16) or it unreliably found a local minimum, but further tests were required to see if it was the desired solution of (3-9). It was the search for effective tests that led to the development of the random search procedure. Thus the conclusion was reached that penalty function techniques and gradient type search procedures are totally inapplicable to complex nonlinear, high order problems.

The futility of using gradient techniques and penalty functions became even more clearly apparent when we considered the 45 parameter search for the optimal Q matrix, Q^o. An analytical gradient could not be derived because ℓ was not explicitly available, and a numerical gradient would require 45 function evaluations (one for each single perturbation of a parameter) to estimate the gradient at one point, and each evaluation would require one search for ℓ. Thus, one could envision $45 \times 20 = 900$ minutes of computer time for one gradient calculation for the Q^o search. [The gradient procedure was using 20 minutes of computing time to converge to a minimum of (3-16), not necessarily the solution of (3-9).] Those 45 function evaluations would be more productive if we allowed for chance and good fortune, i.e., if we used a random search procedure to select evaluation points rather than clustering them about some arbitrary starting point in order to calculate a gradient. This reasoning led us to devote almost all of our remaining effort to the development of more efficient random search procedures.

An efficient random search technique which takes advantage of the geometry of the state space was developed and is discussed fully in Appendix D, part (iv). This search procedure produces results with a high level confidence, does not produce the trivial solution, has no scaling problems, requires no gradient computations, and accomplishes a determination of ℓ in 20 to 30 seconds, depending on how the points fall, i.e., how many actual function evaluations and one dimensional (bisection) searches are carried out. This underscores the value of designing the technique to fit the problem rather than forcing a problem to fit a technique.

This inner search, or search of the state space, has been resolved in an entirely satisfactory way, largely because information is available in the way of geometrical structure in dealing with a collection of nested ellipsoids: The search as finally refined is fast, dependable, and accurate.

The outer search, or search of the 45 dimensional parameter space, which is detailed in Appendix D, part (v), has not been developed to a completely satisfactory point, not only because of the high dimension of the parameter space, but also because almost nothing is known about the geometry of this space, that is, the general nature of the functional dependence of the volume $J(Q)$ on the parameters which generate Q, and because of the primitive state of the art in global search techniques.

This algorithm is the first known instance of one random search imbedded in another. The extremely efficient "inner search" (Fig. 5-1) takes between 20 and 30 seconds on the IBM 360-75 to search the nine dimensional OAO control system phase space for a bound on the volume estimate provided by a specific positive definite matrix Q. The 30-second time is valid if 5000 points are examined whereas the lesser times are due to an "abort" feature. This feature holds the last best inverse volume estimate (vol^{-1*}) and compares it with the vol^{-1} as calculated at each randomly selected point. If vol^{-1} is greater than vol^{-1*} the search is aborted since we are now looking at smaller volume estimates than our last best. Since each Q is generated by 45 independent variables (for the OAO) in the "outer search" (Fig. 5-1), runs of one hour or more on the Grumman IBM 360-75 and the IBM 360-95 at the Institute for Space Studies in New York City were required to obtain estimates of the domain of attraction which represented maximal quadratic estimates with some degree of confidence.

The need for research into effective search techniques is a definite future requirement as is evidenced by the effort put forth in this study. It looks as if general procedures to accomplish this will not suffice and the search technique will be a function of the problem itself or of a general class of problems.

The computer program was developed primarily in FORTRAN IV while some subroutines are in machine language. The machines utilized were the IBM 360 series, models 75 and 95. The entire program consists of approximately 1800 cards. The input consists of between 6 and 16 cards depending on the program option selected, while the printed output consists of the inverse volume (a carry-over from initially using gradient minimization routines) of the optimal estimate of the domain of attraction. There is interim output such as the matrix Q and its parameters, the matrix P, the performance index $\underline{P}$, the value of the Liapunov function V, and its time derivative $\dot{V}$ at the point where ℓ is defined whenever there is an improvement in performance $\underline{P}$.

The details of the computer programs are included in Appendix G. A special form of these programs is included in Appendix F where the Liapunov equation is not solved for P from a positive definite Q but instead a positive definite P is chosen that does not necessarily imply that Q will be positive definite. This program just checks Q for positive definiteness before proceeding. As many as 2400 trial choices of positive definite P-matrices have been made consecutively without a positive definite Q occurring.

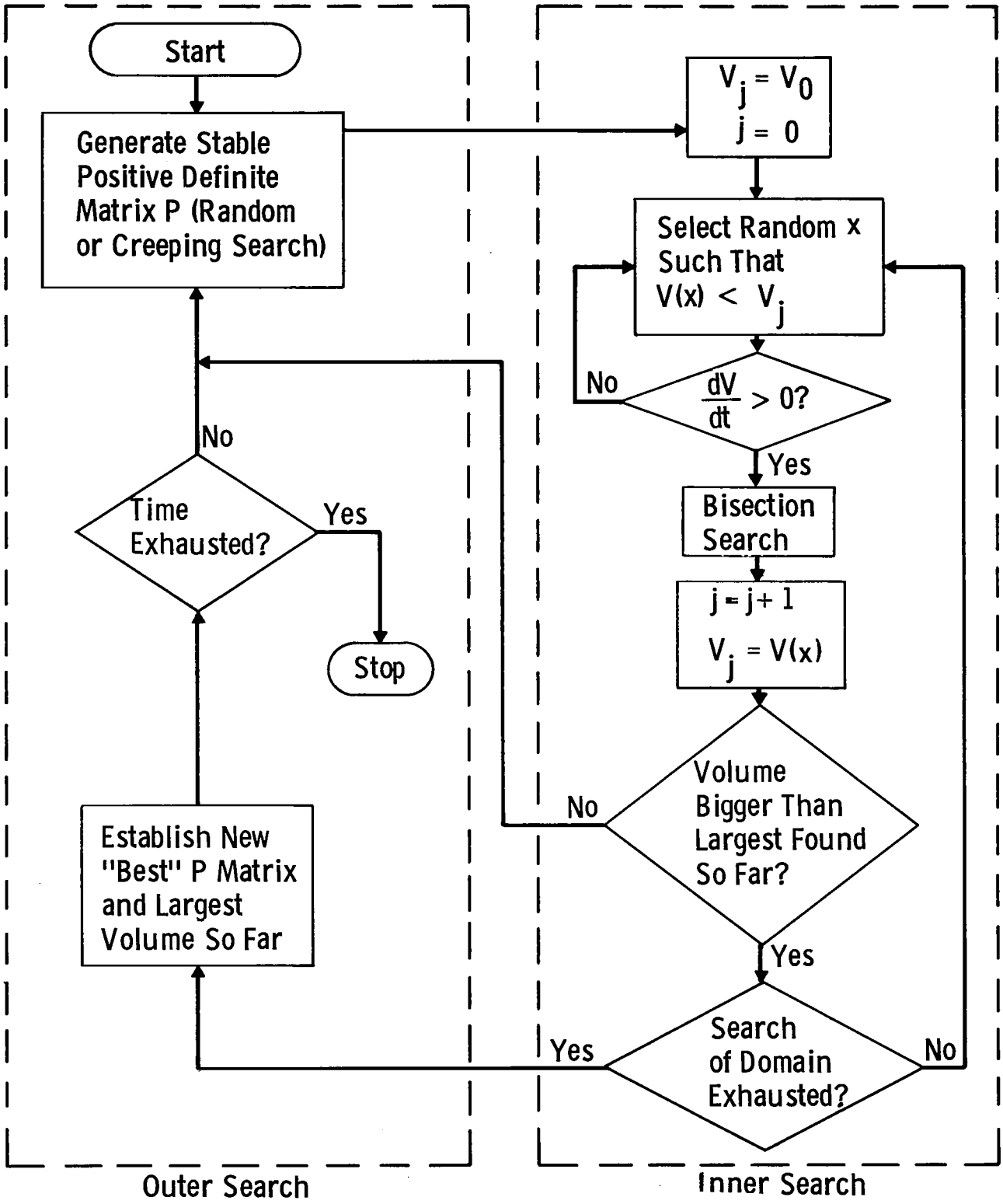


Fig. 5-1 Simplified Flow Chart for the Stability Analysis Algorithm

The experimental results of the program based on selecting the Q matrix are presented for two systems. The first is the nine dimensional complete system, while the second is a six dimensional version of the first with the compensator lag dynamics eliminated because of their much higher rate of response. There is nevertheless a basis of comparison of the two systems in their respective angles and associated momenta.

The over-all result of the computation was that the estimates were well into the nonlinear region of the system. That is, the angular error which causes motor saturation is approximately twenty arc seconds, whereas our results indicated that we were closer to ten minutes of arc in our estimates of the angular state limits of the domain of attraction. While the ten arc minutes looks good with respect to the motor saturation error signal, we observed consistent stable behavior of the system during simulation runs from fifteen degrees of attitude error for many choices of initial star tracker command gimbal angles and initial angular momenta of the system.

We were fortunately able to use the IBM 360/95 at the NASA Goddard Institute for Space Studies (ISS). This computer is much faster (approximately ten times) than the IBM 360/75 which we had been using at Grumman and has a core 10 times larger. This enabled us to use a 300,000 point inner loop state space search, rather than the 5000 point search being used, in order to attain confidence in the validity of the random search technique results. The results of the 9 dimensional search as run at the ISS are shown in Table 5-1. These results are only a small portion of the total results of computer runs at both Grumman and ISS. The best inverse volume in which there was some degree of confidence was 0.488×10^{35} at run #6, which corresponds to physical variable limits (after maximal eigenvector projections on unscaled state coordinates as shown in Appendix E) of: $|\phi|$ = 2.48 min., $|v_\phi|$ = 0.181 ft-lb-sec, $|\omega_\phi|$ = 1980 volts, $|\theta|$ = 5.91 min., $|v_\theta|$ = 0.342 ft-lb-sec, $|\omega_\theta|$ = 941 volts, $|\psi|$ = 8.98 min., $|v_\psi|$ = 0.452 ft-lb-sec, $|\omega_\psi|$ = 1410 volts. It should be noted that the angle intercepts $|\phi|$, $|\theta|$, and $|\psi|$ were diminished by factors of approximately three from those obtained with previous 5000-point searches, due to the more conservative influence of the larger number of points used in the inner-loop search. These results were obtained with a quasi-diagonal Q matrix, zero initial momenta ($Ih_\phi^o = Ih_\theta^o = Ih_\psi^o = 0$), $\sin(\gamma_{1c} - \gamma_{2c}) = 0.1$, $\beta_{1c} = 0$ (quasi-diagonal A), and $\beta_{2c} = -\pi/6$ radians (run #6, Table 5-1).

Table 5-1

TABULATED RESULTS OF Q-MATRIX SEARCH PROGRAM AT THE INSTITUTE FOR SPACE STUDIES

Run #	Trials	Time (Min)	vol^{-1}*	P*	Comments
1*	~ 150	3	$.606 \times 10^{33}$	1.1217	5000-pt. search, quasi-diagonal Q (q-d-Q); job aborted - excessive output
2*	~ 2500	62	(not printed out)	1.1480	5000-pt. search, q-d-Q, reduced output run
3*	1019	105	$.599 \times 10^{35}$	1.1592	300,000-pt. search from the best point of run 2, q-d-Q (all runs from here on are 300,000 pts)
4*	1000	14	$.497 \times 10^{35}$	1.1566	Started from the best point of run 3, full-Q
5*	2000	16	$.892 \times 10^{35}$	1.1650	Continuation of run 4 in random # gen., full-Q
6*	2473	72	$.488 \times 10^{35}$	1.1563	q-d-Q from best point in run 3 (really an extension of 3)
7**	1022	118	$.187 \times 10^{42}$	1.3757	q-d-Q, start from $\lambda_i = 1$, $\theta_j = \phi_k = 0$
8**	1872	136	$.202 \times 10^{41}$	1.3435	Start from the best point in run 3, q-d-Q
9***	29	38	$.134 \times 10^{49}$	1.6042	Start from the best point in run 3, q-d-Q
10***	453	61	$.105 \times 10^{50}$	1.6340	Same as run 9 except full-Q
11****	1000	2	abort condition† 10^{51}	1.6666	q-d-Q, started from the best point in run 3
12*****	1000	3	abort condition† 10^{51}	1.6666	q-d-Q, started from the best point in run 3
Total Trials	14518				

* $h_i = 0$, $\gamma_{1c} = -\gamma_{2c} = .05017822$ rad, $\beta_{1c} = 0$, $\beta_{2c} = -\pi/6$

** $h_i = 0$, $\gamma_{1c} = -\gamma_{2c} = \pi/4$, $\beta_{1c} = 0$, $\beta_{2c} = -\pi/6$

*** $h_i = 0$, $\gamma_{1c} = -\gamma_{2c} = \pi/4$, $\beta_{1c} = -\beta_{2c} = \pi/6$

**** h_i = 1/1500 (half wheel speed)+,-,+, $\gamma_{1c} = -\gamma_{2c} = .05017822$, $\beta_{1c} = 0$, $\beta_{2c} = -\pi/6$

***** h_i = 0.2/1500 (1/10 wheel speed)+,+,+, $\gamma_{1c} = -\gamma_{2c} = .050178$, $\beta_{1c} = 0$, $\beta_{2c} = -\pi/6$

† if the best Liapunov function is $< 10^{-12}$, the vol^{-1} is set = 10^{51} and P* thus becomes ~ 5/3

The inclusion of a full Q-matrix, addition of nonzero initial momenta, the inclusion of $\beta_{1c} \neq 0$, and increasing the $\sin(\gamma_{1c} - \gamma_{2c})$ to values > 0.1 cause degradation of the vol^{-1} as illustrated in Table 5-1. Results indicate high sensitivity to variations in γ_{1c}, γ_{2c}, β_{1c}, and initial momenta. The case of $\beta_{1c} = 0$ gives a quasi-diagonal A matrix, thereby decoupling the system, at least in the linear part. The nonlinear part is still coupled through the roll, pitch, and yaw channels. The most severe degradation of the system occurred when initial momenta were introduced to even one-tenth of wheel capacity.

A 6 dimensional approximation of the 9 dimensional problem was also programmed but with a 100,000 random point inner loop search. The best results were obtained for the case $\beta_{1c} = 0$, $\beta_{2c} = -\pi/6$ rad, $\sin(\gamma_{1c} - \gamma_{2c}) = 0.1$, $h_\phi^o = h_\theta^o = h_\psi^o = 0$, giving a vol^{-1} estimate $= 0.562 \times 10^{25}$, with projected maximum values (not occurring simultaneously) of the physical variables as follows: $|\phi| = 0.14$ min., $|\theta| = 10.4$ min., $|\psi| = 8.30$ min., $|v_\phi| = 0.050$ lb-ft-sec, $|v_\theta| = 0.48$ lb-ft-sec, $|v_\psi| = 0.39$ lb-ft-sec (see Table 5-2).

A comparison of the 6 and 9 dimensional results show that both programs exhibited their best results for the case of 1) quasi-diagonal Q-matrix, 2) zero initial momenta ($Ih_\phi^o = Ih_\theta^o = Ih_\psi^o = 0$), 3) $\sin(\gamma_{1c} - \gamma_{2c}) = 0.1$, 4) $\beta_{1c} = 0$, and 5) $\beta_{2c} = -\pi/6$ rad. The 6 dimensional results provided a better $|\theta|$ intercept estimate of 10.4 min. than the 9 dimensional estimate of 5.91 min.; however, the 9 dimensional program provided surprisingly better $|\phi|$ and $|\psi|$ estimates (2.48 min. compared to 0.14 min. for $|\phi|$, and 8.98 min. compared to 8.30 min. for $|\psi|$). Perhaps the failure of the 6 dimensional program to provide clearly superior estimates in spite of the smaller dimension of its Q parameter search and greater relative number of trials is due to the fact that the effect of a 100,000 point search per trial in 6 dimensions is approximately equivalent to a 32 million point search in 9 dimensions; 32 million would be very conservative compared with the 300,000 point search actually used in the 9 dimensional case.

The dollar cost of obtaining an estimate of the domain of attraction is of interest for comparison with simulation. Only nominal values of the cost per computer usage hour are used since these costs fluctuate as a function of total usage hours, time of day, order of priority, etc. The cost of obtaining an estimate of

Table 5-2

SUMMARY OF RUNS AT ISS FOR 6 DIMENSIONAL MODEL

Run #	Trials	Time (Min)	vol^{-1}	P^*	Comments
1*	~ 8000	120	$.562 \times 10^{25}$	1.237	100,000 point search, $P^* = \log(vol)^{-1}/20$
2***	~ 3000	60	$.261 \times 10^{30}$	1.471	" "
3**	4200	60	$.225 \times 10^{27}$	1.318	" "
4****	4000	60	----	---	vol^{-1} too small to compute without rescaling problem

* $h^o = 0$, $\gamma_{1c} = -\gamma_{2c} = .05017822$ rad, $\beta_{1c} = 0$, $\beta_{2c} = -\pi/6$ rad

*** $h^o = 0$, $\gamma_{1c} = -\gamma_{2c} = .05017822$ rad, $\beta_{1c} = \pi/6$, $\beta_{2c} = -\pi/6$ rad

** $h^o = 0$, $\gamma_{1c} = -\gamma_{2c} = \pi/4$, $\beta_{1c} = 0$, $\beta_{2c} = -0/6$ rad

**** $h^o = 1/1500$ (+,-,+), $\gamma_{1c} = -\gamma_{2c} = .05017822$ rad, $\beta_{1c} = 0$, $\beta_{2c} = -\pi/6$ rad

the domain of attraction is found for a given set of initial momenta ($h_i^o = 0$) and a particular set of commanded gimbal angles ($\gamma_{1c} = -\gamma_{2c} = 0.05017822$ rad, $\beta_{1c} = 0$ and $\beta_{2c} = -\pi/6$). These data are presented as runs #1 through #6 in Table 5-1. The best estimate (run #6) is used as the example, (it must be noted that run #6 is a continuation of run #3 which in turn is a continuation of run #2). Therefore the cost should be the dollar value associated with the sum total of computer hours used, which is approximately four. Since the IBM 360/95 nominally costs $1000/computer usage hour, the cost is approximately $4000/estimate. Since we do not want it to appear that any issues are being clouded by the particular parameter set mentioned, and since this is possibly not the usual computer, we adjust our estimate by a factor of ten to the conservative side, as a result of which our cost is brought up to $40,000/estimate/parameter set. Let us get some "feel" for the

number of points in the state space that have been examined during these four hours of computer time. With the special abort feature of the program, which reinitializes the state space search whenever the volume of the estimate corresponding to Liapunov function V becomes smaller than the previous last best volume estimate, a 300,000 point state space search takes approximately five minutes. If we say that for only half the time we are examining 300,000 point/5 minutes and the other time we are doing nothing, then we are evaluating 72×10^5 points in this over-all search. Hence our estimate is approximately 0.5¢/point/parameter set.

Since stability analysis by simulation is the major means of doing the same job that this algorithm is doing, a comparison might be undertaken to get a dollar estimate of the cost of doing such an analysis with a satisfactory degree of confidence in the results. Since we have a nine dimensional state space, choosing all combinations of the maximum value, minimum value, and zero for each of the state variables would yield a hypercube grid of $3^9 - 1$ (excluding the origin) state points (initial conditions from which trajectories should be run). Again, this is for a single set of momenta and commanded gimbal angle parameters. On the average, a trajectory in state space takes approximately three to five minutes of IBM 360/75 machine time. If we estimate the IBM 360/75 machine time cost to be $500/computer usage hour, the cost to run this simulation from which stability of the system is to be ascertained and in which we think one might place a high degree of confidence is $500,000, which is $25/point/parameter set. In these estimates we have taken $3^9 - 1$ to be 20,000, we have assumed the three minute/run figure, and we have ignored the cost of plotting and "eye balling" the runs.

While it is admittedly true that performance information results from simulation, the problem at hand is stability analysis of the system and not the accumulation of other valid but superfluous data.

As an example of the time and dollar constraints that are placed on a project and how they manifest themselves in what we believe to be a reduction in confidence in the results in terms of stability analysis, consider the following. For the actual OAO, the total number of simulation runs with variation in all parameters was 20,000. Recall that 20,000 was what we believed would be a reasonable number of runs for a single parameter set. Hence the degree of confidence in the results must be considered small.

A conclusion from this experimental work is that the algorithm is a feasible method for conducting the stability analysis of a high order system. It is cost effective compared to simulation, but it is probably too conservative in its estimates although it did give results well into the nonlinear region of operation of the system. Finally, it is clear to us that more efficient search techniques are a requirement for any future work in this area since this was actually the biggest restraint on our over-all progress during this study

6. CONCLUSIONS

The algorithm developed and examined in this study is a feasible solution to the problem of estimating the domain of attraction of an equilibrium state of a complex nonlinear physical system. Its further development into a practical tool is limited, however, in two ways.

First, there is the problem of determining the subset of systems within the set of quasilinear asymptotically stable systems for which a quadratic or ellipsoidal estimate of the domain of attraction is adequate. Since the nonlinearities are not accounted for in the geometric shape of the estimate it is conceivable that there are systems for which one cannot find a suitably large family of ellipsoids within the domain of attraction such that all trajectories cut them in the inward direction as required by the problem formulation. The use of Luré-Liapunov functions may be more suitable in these cases, but the computational problems become more difficult.

Second, the state of the art in search techniques must be substantially advanced so that the maximization problem can be effectively and efficiently solved. For a system of dimension n the maximization problem is of dimension $n(n + 1)/2$. Thus, for problems of reasonable complexity gradient techniques must be abandoned in favor of random search techniques because more information can be gathered, in a global sense, by the same effort required to determine a local piece of information (the gradient). The problem is further complicated because the objective function is stated in terms of the solution of a minimum problem which is dependent upon the variables over which the maximization occurs. Thus the problem is one of high dimension and unknown, complex geometry.

Despite these hurdles, the continued development of the algorithm remains attractive because it promises to provide a tool that can provide higher confidence stability information for complex nonlinear systems than can simulation (the only other similarly general tool) at lower cost. This conclusion is based on a very conservative comparison, i.e., all efforts were made to favor simulation. For a given confidence level the cost of obtaining the result by simulation is two orders of magnitude more expensive than the cost of utilizing the algorithm.

In the process of developing the algorithm, the inapplicability of gradient searches and penalty function techniques to the minimum

problem became painfully clear. They are inapplicable for three reasons: 1) the solution sought is the local minimum nearest in value to the global minimum (a trivial solution); 2) the surface being searched has many minima; and 3) the relative ranges of the function to be minimized and the penalty function are not known a priori. The global minimum acts as a grand attractor and any attempt to "mask it from view" introduces a new unknown local minimum or requires total knowledge of the surface around the origin. The many local minima repeatedly trap a gradient search and the result must be certified by other means. The lack of knowledge about the relative ranges of the two functions makes the choice of penalty constant almost impossible without excessive additional computation equivalent to random searching. This factor and the need for another means of certification led to creation of a very effective random search technique that utilized the known geometry. The resulting conclusion is that for complex problems, random search is more attractive and effective than gradient search, and, if at all possible, known geometric facts should be used in structuring the search.

In the process of developing the system model it was shown that the choice of star tracker model directly affects the complexity of the stability analysis. Further, linearization of the star tracker-error processor combination introduces substantial errors in both magnitude and sign of the approximate error signals. This is apprently the reason why the Popov approximation (linear part plus saturation) fails to represent the stability properties of the system. (It indicates stability when the system is unstable.) The lesson learned is that one must be careful in modeling a complex nonlinear system and must resist the temptation of settling for an analytically attractive approximate model until it has been proved adequate to the task. It is not at all clear that the adequacy can be proved without analyzing both the complete model and the approximation, thus negating the value of the approximation.

The Popov approximation was studied in connection with the use of Luré-Liapunov functions to obtain an improved estimate of the domain of attraction. It was found that the frequency domain techniques are not sufficiently well developed to permit the complete stability analysis of the approximate model. Only the special case in which the three channels are uncoupled could be carried out. It also became apparent that computational aids are required to establish the positivity properties for the modified system functions when the system is nonacademic, i.e., as complex as the one studied here.

In the formulation of the Popov model more respect was gained for the prosaic but ubiquitous block diagram. Formulation of the model from the block diagram is easy compared to the laborious task of formulating it from the state equations. The lesson here is not to follow the deceptively simple prescription of matrix manipulations until the engineer's intuition has first been exercised. It has a place even in the world of high order nonlinear state equations.

APPENDIX A

DERIVATION OF THE SYSTEM MODEL AND APPROXIMATIONS

The OAO is a cylindrical structure of octagonal cross section designed to accommodate a wide variety of astronomical experiment packages. This attitude control system has four modes of operation: initial stabilization, acquisition of the sun line and then roll search to acquire the guide stars; open loop slew to a commanded attitude; coarse pointing control; and fine pointing control. This appendix is concerned with the coarse pointing mode, which utilizes a high gain nonlinear system capable of reducing initial attitude errors of 8° to a range of 2 to 5 minutes of arc such that the fine pointing system can hold the required attitude to ± 15 seconds of arc for up to 50 minutes. (See [13, 14] for details.) High system gain and nonlinearities combine to make the determination of coarse pointing stability very difficult.

Further, instability of the coarse pointing mode can destroy success of the mission. Since observations and corrections can only be made at three closely spaced ground stations, an instability can go unobserved for almost an hour. A tumbling instability will cause the control system activity to deplete the stored electrical energy. It will also prevent the solar cell arrays from recharging the batteries and cause the trackers to hit the gimbal stops thus losing the references and attitude information. The remainder of this presentation is devoted to developing the system model for the difficult and important problem of determining the stability of the coarse pointing mode. The system block diagram of the coarse pointing mode is shown in Fig. A-1 with the signals at each block of the three channel system explicitly identified as a three vector.

The primary sources of mechanical energy in the coarse pointing mode are electrically driven momentum wheels. The primary sensors in the coarse pointing attitude control system are a set of gimbaled star trackers that track selected celestial references (guide stars) and read out gimbal angle errors. These errors are the differences between the gimbal angles when the vehicle is on target (commanded gimbal angles) and the actual gimbal angles required to maintain the guide star in the center of the star tracker's field of view.

The actual OAO has six gimbaled star trackers and uses from two to six star trackers at a time; however, the "paired-tracker" system devised by Doolin and Showman [1, 2] has only the four star trackers mounted on the sides of the vehicle and uses a pair at a time. The simplification in error processing and system design/analysis of the "paired-tracker" system is achieved by aligning the outer gimbal axes of all four trackers with the vehicle optical or roll axis. This is the system that will be modeled here.

Our attention in this appendix is directed toward the derivation of the system model. It will be demonstrated that the choice of the model has a direct material effect on the degree of difficulty of the stability analysis. The basic block diagram of the system is given in Fig. A-1 and the model will be developed in signal flow sequence beginning with the star trackers.

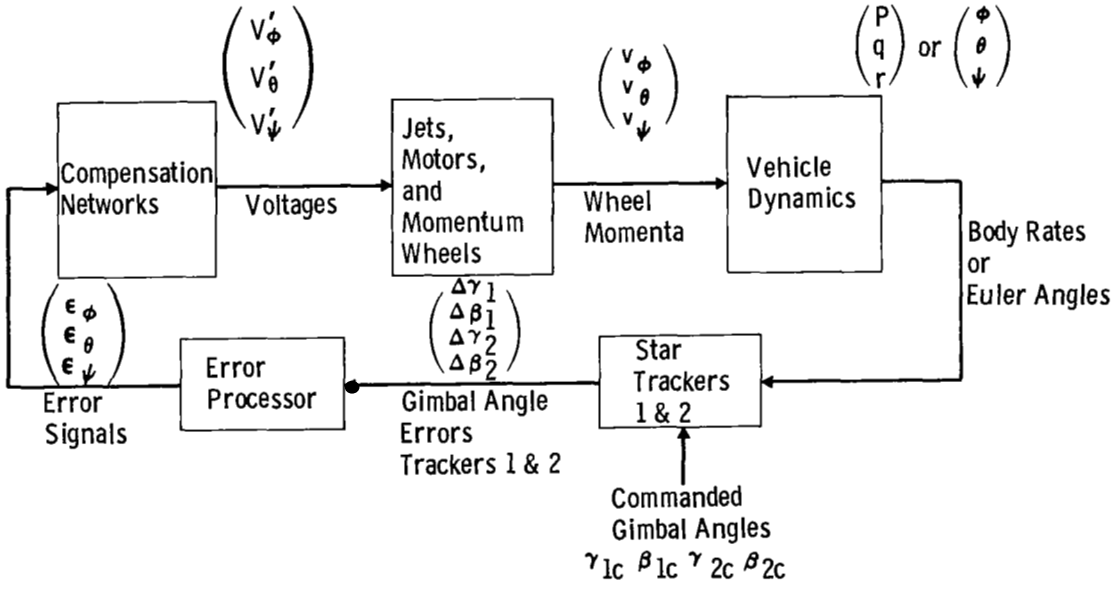


Fig. A-1 Basic System Block Diagram

To define the star tracker model we begin by establishing the coordinate system of Fig. A-2. The coordinate system subscript r (reference) is assumed to be inertially fixed with the X_r coordinate axis being the line of sight to the target star, and the Y_r, Z_r axes are in a plane perpendicular to the line of sight to the target star such that the angles ϕ, θ, ψ are zero when the vehicle is on target. The coordinate system subscript b (body) is aligned with the principal inertia axis (control axes) of the vehicle, with the X_b axis being the vehicle optical axis. The Euler angles ϕ, θ, ψ are, respectively, the roll, pitch, and yaw angles with respect to the reference coordinates. (The convention here is that vectors denoting coordinate frames are upper case, components of a vector lower case.) Thus the relationship of the reference and body coordinates is given by the set of rotation transformations R_ϕ, R_θ, R_ψ, viz.,

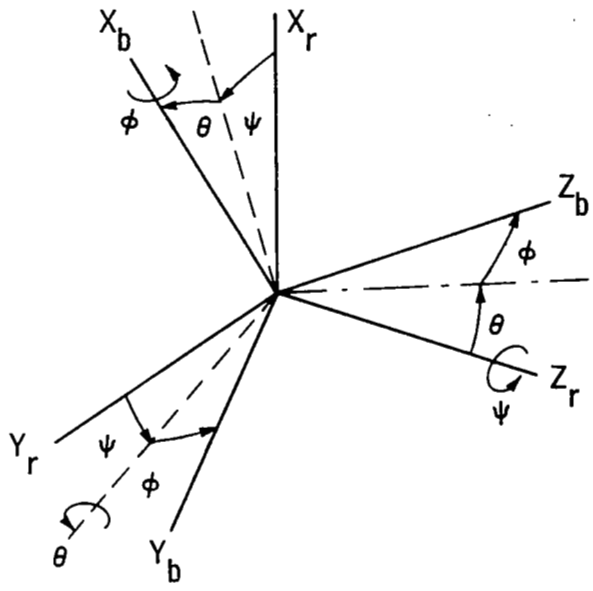
$$\begin{bmatrix} X_r \\ Y_r \\ Z_r \end{bmatrix} = R_\psi R_\theta R_\phi \begin{bmatrix} X_b \\ Y_b \\ Z_b \end{bmatrix} \tag{A-1}$$

where

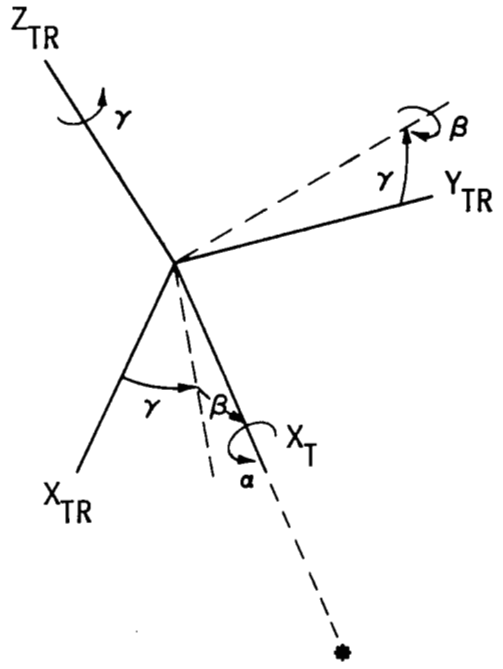
$$R_\phi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & -s\phi \\ 0 & s\phi & c\phi \end{bmatrix}$$

$$R_\theta = \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix} \tag{A-2}$$

$$R_\psi = \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



b - body coordinates
r - reference coordinates
ϕ - roll
θ - pitch
ψ - yaw



TR - TRACKER REFERENCE
T - TRACKER OPTICAL AXIS
β - INNER GIMBAL ANGLE
γ - OUTER GIMBAL ANGLE

Fig. A-2 Definition of Coordinate Systems

For each star tracker we construct a coordinate system subscript TR (tracker reference) in which Z_{TR} is aligned with the tracker outer gimbal axis, Y_{TR} is aligned with the nominal inner gimbal axis, and X_{TR} is aligned with the nominal tracker telescope axis. The coordinate X_T is the actual tracker optical axis, and the angles γ, β are the outer and inner gimbal angles. The angle α is the rotation about the tracker optical axis. Thus the relationship between the tracker and the tracker reference coordinates is given by the rotation transformations R_α, R_β, R_γ, viz.,

$$\begin{bmatrix} X_T \\ Y_T \\ Z_T \end{bmatrix} = R_\alpha R_\beta R_\gamma \begin{bmatrix} X_{TR} \\ Y_{TR} \\ Z_{TR} \end{bmatrix} \tag{A-3}$$

where

$$R_\alpha = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha & s\alpha \\ 0 & -s\alpha & ca \end{bmatrix}$$

$$R_\beta = \begin{bmatrix} c\beta & 0 & -s\beta \\ 0 & 1 & 0 \\ s\beta & 0 & c\beta \end{bmatrix} \tag{A-4}$$

$$R_\gamma = \begin{bmatrix} c\gamma & s\gamma & 0 \\ -s\gamma & c\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The relationship of a given tracker (no. n) reference coordinate system to the vehicle coordinate system is given by the linear transformation T_n, i.e.,

$$\begin{bmatrix} X_b \\ Y_b \\ Z_b \end{bmatrix} = T_n \begin{bmatrix} X_{TR} \\ Y_{TR} \\ Z_{TR} \end{bmatrix} \qquad \text{(A-5)}$$

where

$$T_1 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$T_2 = \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \qquad \text{(A-6)}$$

$$T_3 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$T_4 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

The star tracker model will now be derived by assuming perfect tracking and infinite distance from the stars. In this case, with $\phi = \theta = \psi = 0$, the commanded gimbal angles give the coordinates of the unit vector along the line of sight to the guide star in the reference coordinates, i.e.,

$$\begin{bmatrix} \tilde{X}_r \\ \tilde{Y}_r \\ \tilde{Z}_r \end{bmatrix} = T_n R_{\gamma_c}^T R_{\beta_c}^T R_{\alpha_c}^T \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{A-7}$$

where superscript T denotes transpose. However, if the body undergoes a rotation of ϕ, θ, ψ then the actual gimbal angles describe the line of sight in body coordinates, i.e.,

$$\begin{bmatrix} \tilde{X}_b \\ \tilde{Y}_b \\ \tilde{Z}_b \end{bmatrix} = T_n R_\gamma^T R_\beta^T R_\alpha^T \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{A-8}$$

and its description in reference coordinates is

$$\begin{bmatrix} \tilde{X}_r \\ \tilde{Y}_r \\ \tilde{Z}_r \end{bmatrix} = R_\psi R_\theta R_\phi T_n R_\gamma^T R_\beta^T R_\alpha^T \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{A-9}$$

Since the line of sight to the star is fixed in inertial coordinates we have

$$T_n R_{\gamma_c}^T R_{\beta_c}^T R_{\alpha_c}^T \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = R_\psi R_\theta R_\phi T_n R_\gamma^T R_\beta^T R_\alpha^T \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{A-10}$$

From this set of three equations we obtain two equations relating the actual and commanded values of β and γ. The error $\Delta\beta$, $\Delta\gamma$ is defined to be the difference between corresponding actual and

commanded values. In the notation of Doolin and Showman [1, 2], the angles β and γ are defined to be the negatives of those defined here. With this taken into account, the equations defining $\Delta\beta$, $\Delta\gamma$ for the four trackers are given in Fig. A-3b.

The tracker model can also be derived by noting that the rotational rate of the body with respect to inertial space in body coordinates is given by

$$\Omega_b = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{A-11}$$

where p, q, r are the rotational rates about the body axes X_b, Y_b, Z_b. Thus the rotational rate in reference coordinates is given by

$$\Omega_r = R_\psi R_\theta R_\phi \Omega_b \ . \tag{A-12}$$

However, the tracker coordinate system is also inertially fixed and the rotational rate of the body in that system is given by

$$\Omega_{TR} = - \begin{bmatrix} \dot{\alpha}\, c\beta c\gamma - \dot{\beta}\, s\gamma \\ \dot{\beta}\, c\gamma + \dot{\alpha}\, c\beta s\gamma \\ \dot{\gamma} - \dot{\alpha}\, s\beta \end{bmatrix} = - \begin{bmatrix} c\beta c\gamma & -s\gamma & 0 \\ c\beta s\gamma & c\gamma & 0 \\ -s\beta & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} = - T_{\beta\gamma} \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} \tag{A-13}$$

in the tracker reference system. In the reference system this is given by

$$\Omega_r = R_\psi R_\theta R_\phi T_n \Omega_{TR} \tag{A-14}$$

and thus

$$R_\psi R_\theta R_\phi \Omega_b = R_\psi R_\theta R_\phi T_n \Omega_{TR} \tag{A-15}$$

$$\dot{\Delta\beta}_1 = s(\Delta\gamma_{1C} + \gamma_{1C}) \cdot q + c(\Delta\gamma_{1C} + \gamma_{1C}) \cdot r$$

$$\dot{\Delta\gamma}_1 = p - t(\Delta\beta_1 + \beta_{1C}) \cdot c(\Delta\gamma_{1C} + \gamma_{1C}) \cdot q + t(\Delta\beta_1 + \beta_{1C}) \cdot s(\Delta\gamma_{1C} + \gamma_{1C}) \cdot r$$

$$\dot{\Delta\beta}_2 = -s(\Delta\gamma_2 + \gamma_{2C}) \cdot q - c(\Delta\gamma_2 + \gamma_{2C}) \cdot r$$

$$\dot{\Delta\gamma}_2 = p + t(\Delta\beta_2 + \beta_{2C}) \cdot c(\Delta\gamma_2 + \gamma_{2C}) \cdot q - t(\Delta\beta_2 + \beta_{2C}) \cdot s(\Delta\gamma_2 + \gamma_{2C}) \cdot r$$

$$\dot{\Delta\beta}_3 = -c(\Delta\gamma_3 + \gamma_{3C}) \cdot q + s(\Delta\gamma_3 + \gamma_{3C}) \cdot r$$

$$\dot{\Delta\gamma}_3 = p - t(\Delta\beta_3 + \beta_{3C}) \cdot s(\Delta\gamma_3 + \gamma_{3C}) \cdot q - t(\Delta\beta_3 + \beta_{3C}) \cdot c(\Delta\gamma_3 + \gamma_{3C}) \cdot r$$

$$\dot{\Delta\beta}_4 = c(\Delta\gamma_4 + \gamma_{4C}) \cdot q - s(\Delta\gamma_4 + \gamma_{4C}) \cdot r$$

$$\dot{\Delta\gamma}_4 = p + t(\Delta\beta_4 + \beta_{4C}) \cdot s(\Delta\gamma_4 + \gamma_{4C}) \cdot q + t(\Delta\beta_4 + \beta_{4C}) \cdot c(\Delta\gamma_4 + \gamma_{4C}) \cdot r$$

a) GIMBAL ANGLE RATE EQUATIONS

$$\Delta\beta_1 = \sin^{-1}(c\psi\, c\theta\, s\beta_{1C} + s\psi\, c\theta\, c\gamma_{1C}\, c\beta_{1C} + s\theta\, s\gamma_{1C}\, c\beta_{1C}) - \beta_{1C}$$

$$\Delta\gamma_1 = \tan^{-1}\left(\frac{-(s\psi s\phi + c\psi s\theta c\phi)\, s\beta_{1C} + (c\psi s\phi - s\psi s\theta c\phi)\, c\gamma_{1C}\, c\beta_{1C} + c\theta c\phi s\gamma_{1C}\, c\beta_{1C}}{-(s\psi c\phi - c\psi s\theta s\phi)\, s\beta_{1C} + (c\psi c\phi + s\psi s\theta s\phi)\, c\gamma_{1C}\, c\beta_{1C} - c\theta s\phi s\gamma_{1C}\, c\beta_{1C}}\right) - \gamma_{1C}$$

$$\Delta\beta_2 = \sin^{-1}(c\psi\, c\theta\, s\beta_{2C} - s\psi\, c\theta\, c\gamma_{2C}\, c\beta_{2C} - s\theta\, s\gamma_{2C}\, c\beta_{2C}) - \beta_{2C}$$

$$\Delta\gamma_2 = \tan^{-1}\left(\frac{(s\psi s\phi + c\psi s\theta c\phi)\, s\beta_{2C} + (c\psi s\phi - s\psi s\theta c\phi)\, c\gamma_{2C}\, c\beta_{2C} + c\theta c\phi s\gamma_{2C}\, c\beta_{2C}}{(s\psi c\phi - c\psi s\theta s\phi)\, s\beta_{2C} + (c\psi c\phi + s\psi s\theta s\phi)\, c\gamma_{2C}\, c\beta_{2C} - c\theta s\phi s\gamma_{2C}\, c\beta_{2C}}\right) - \gamma_{2C}$$

$$\Delta\beta_3 = \sin^{-1}(c\psi\, c\theta\, s\beta_{3C} + s\psi\, c\theta\, s\gamma_{3C}\, c\beta_{3C} - s\theta\, c\gamma_{3C}\, c\beta_{3C}) - \beta_{3C}$$

$$\Delta\gamma_3 = \tan^{-1}\left(\frac{-(s\psi c\phi - c\psi s\theta s\phi)\, s\beta_{3C} + (c\psi c\phi + s\psi s\theta s\phi)\, s\gamma_{3C}\, c\beta_{3C} + c\theta s\phi c\gamma_{3C}\, c\beta_{3C}}{(s\psi s\phi + c\psi s\theta c\phi)\, s\beta_{3C} - (c\psi s\phi - s\psi s\theta c\phi)\, s\gamma_{3C}\, c\beta_{3C} + c\theta c\phi c\gamma_{3C}\, c\beta_{3C}}\right) - \gamma_{3C}$$

$$\Delta\beta_4 = \sin^{-1}(c\psi\, c\theta\, s\beta_{4C} - s\psi\, c\theta\, s\gamma_{4C}\, c\beta_{4C} + s\theta\, c\gamma_{4C}\, c\beta_{4C}) - \beta_{4C}$$

$$\Delta\gamma_4 = \tan^{-1}\left(\frac{+(s\psi c\phi - c\psi s\theta s\phi)\, s\beta_{4C} + (c\psi c\phi + s\psi s\theta s\phi)\, s\gamma_{4C}\, c\beta_{4C} + c\theta s\phi c\gamma_{4C}\, c\beta_{4C}}{-(s\psi s\phi + c\psi s\theta c\phi)\, s\beta_{4C} - (c\psi s\phi - s\psi s\theta c\phi)\, s\gamma_{4C}\, c\beta_{4C} + c\theta c\phi c\gamma_{4C}\, c\beta_{4C}}\right) - \gamma_{4C}$$

b) GIMBAL ANGLE EQUATIONS

Fig. A-3 Two Possible Star Tracker Models

or

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = - T_n T_{\beta\gamma} \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} \quad . \tag{A-16}$$

With the Doolin and Showman convention taken into account, the relations for each tracker are given in Fig. A-3a. Note that α is unmeasurable and thus of no concern here.

Thus, there are two readily derived models for the star trackers — a model based on rate equations and a model based on angle equations. A very tedious exercise in algebra and calculus proves that the former is the derivative of the latter — as it must be. Their influence on the stability analysis will be demonstrated later.

The control system design described in [1] and [2] is given for both a "constant processor" and a "partial processor." These processors differ in the degree to which they approximate the relationship between the tracker gimbal angle errors and the body angle errors. The "partial processor" requires resolvers mounted on the outer gimbal shaft and is the one to be used in this model. For the particular case of trackers 1 and 2 the "partial processor" is given by

$$\begin{bmatrix} \epsilon_\phi \\ \epsilon_\theta \\ \epsilon_\psi \end{bmatrix} = \begin{bmatrix} a_{11} & 1 & a_{13} \\ d_{12}\, c(\gamma_{2c} + \Delta\gamma_2) & 0 & d_{12}\, c(\gamma_{1c} + \Delta\gamma_1) \\ -d_{12}\, s(\gamma_{2c} + \Delta\gamma_2) & 0 & -d_{12}\, s(\gamma_{1c} + \Delta\gamma_1) \end{bmatrix} \begin{bmatrix} \Delta\beta_1 \\ \Delta\gamma_1 \\ \Delta\beta_2 \end{bmatrix} \tag{A-17}$$

where ϵ_ϕ, ϵ_θ, ϵ_ψ are the error signals in the ϕ, θ, ψ channels, respectively. Each of these signals is then passed through a lead-lag compensation network with transfer function $G_c(s)$ where

$$G_c(s) = K_c \frac{(\tau_1 + \tau_2)s + 1}{\tau_2 s + 1}$$

$$K_c = 2.685 \cdot 10^5 \text{ volt/rad} \qquad \text{(A-18)}$$

$$\tau_1 = 4.5 \text{ sec}, \quad \tau_2 = 0.5 \text{ sec}$$

as given in [2]. The corresponding differential equation is

$$\tau_2 \dot{V} + V' = K_c(\tau_1 + \tau_2)\dot{\epsilon} + K_c\epsilon \qquad \text{(A-19)}$$

where $(\dot{})$ denotes differentiation with respect to time and V' is the output voltage. By defining

$$V' = \omega + K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon \qquad \text{(A-20)}$$

Eq. (A-19) becomes

$$\dot{\omega} + \frac{1}{\tau_2}\omega = -K_c \frac{\frac{\tau_1}{\tau_2}}{\tau_2}\epsilon \qquad \text{(A-21)}$$

and the set of equations, in state variable form, that describe the compensation networks are:

$$\begin{aligned}
\dot{\omega}_\phi + \frac{1}{\tau_2}\omega_\phi &= -K_c \frac{\frac{\tau_1}{\tau_2}}{\tau_2}\epsilon_\phi \\
V'_\phi &= \omega_\phi + K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon_\phi \\
\dot{\omega}_\theta + \frac{1}{\tau_2}\omega_\theta &= -K_c \frac{\frac{\tau_1}{\tau_2}}{\tau_2}\epsilon_\theta \\
V'_\theta &= \omega_\theta + K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon_\theta \\
\dot{\omega}_\psi + \frac{1}{\tau_2}\omega_\psi &= -K_c \frac{\frac{\tau_1}{\tau_2}}{\tau_2}\epsilon_\psi \\
V'_\psi &= \omega_\psi + K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon_\psi
\end{aligned} \qquad \text{(A-22)}$$

As per [2] the motors and momentum wheels are represented by the motor saturation

$$V'' = f(V') \tag{A-23}$$

where V'', in volts, is

$$V'' = f(V') = \begin{cases} 26 & , \quad V' > 26 \text{ volts} \\ V' & , \quad |V'| \leq 26 \text{ volts} \\ -26 & , \quad V' < -26 \text{ volts} \end{cases} \tag{A-24}$$

and the transfer function from V'' to wheel torque H_w is

$$G_m(s) = \frac{K_m s}{\tau_m s + 1} \tag{A-25}$$

$$K_m = \frac{1}{13} \frac{\text{ft-lb-sec}}{\text{volt}} \quad , \quad \tau_m = 76.8 \text{ sec} .$$

Since the wheel momentum is given by

$$\dot{v} = H_w \tag{A-26}$$

the transfer function from V'' to wheel momentum v is

$$G'_m(s) = \frac{K_m}{\tau_m s + 1} . \tag{A-27}$$

The corresponding differential equation is

$$\tau_m \dot{v} + v = K_m V'' . \tag{A-28}$$

The equations describing the vehicle are:

$$\begin{aligned} I\dot{p} &= -H_{w_\phi} + (rv_\theta - qv_\psi) \\ I\dot{q} &= -H_{w_\theta} + (pv_\psi - rv_\phi) \\ I\dot{r} &= -H_{w_\psi} + (qv_\phi - pv_\theta) \\ I &= 1500 \text{ slug-ft}^2 \end{aligned} \tag{A-29}$$

where I is the rotational inertia of the vehicle (assumed to be balanced). Thus the equations describing the motors, momentum wheels, and vehicle in state variable form are:

$$\dot{v}_\phi + \frac{1}{\tau_m} v_\phi = \frac{K_m}{\tau_m} V''_\phi$$

$$\dot{p} = + \frac{1}{\tau_m I} v_\phi - \frac{K_m}{\tau_m I} V''_\phi + \frac{1}{I}(rv_\theta - qv_\psi)$$

$$\dot{v}_\theta + \frac{1}{\tau_m} v_\theta = \frac{K_m}{\tau_m} V''_\theta \qquad \text{(A-30)}$$

$$\dot{q} = + \frac{1}{\tau_m I} v_\theta - \frac{K_m}{\tau_m I} V''_\theta + \frac{1}{I}(pv_\psi - rv_\phi)$$

$$\dot{v}_\psi + \frac{1}{\tau_m} v_\psi = \frac{K_m}{\tau_m} V''_\psi$$

$$\dot{r} = + \frac{1}{\tau_m I} v_\psi - \frac{K_m}{\tau_m I} V''_\psi + \frac{1}{I}(qv_\phi - pv_\theta) \ .$$

If the gyroscopic torques due to the momentum wheels are neglected (this appears reasonable on the basis of simulation data) then the vehicle equations reduce to

$$I\dot{p} = - H_{w_\phi} = - \dot{v}_\phi$$

$$I\dot{q} = - H_{w_\theta} = - \dot{v}_\theta \qquad \text{(A-31)}$$

$$I\dot{r} = - H_{w_\psi} = - \dot{v}_\psi$$

or to

$$
\begin{aligned}
Ip &= - v_{\phi} + \left(Ip(0) + v_{\phi}(0) \right) \\
Iq &= - v_{\theta} + \left(Iq(0) + v_{\theta}(0) \right) \\
Ir &= - v_{\psi} + \left(Ir(0) + v_{\psi}(0) \right)
\end{aligned}
\qquad \text{(A-32)}
$$

where $p(0)$ is the initial value of p. These simplified equations clearly demonstrate the momentum exchange process. Using these simplified equations, we reduce the state equations for the momentum wheels and vehicle to:

$$
\begin{aligned}
\dot{v}_{\phi} + \frac{1}{\tau_m} v_{\phi} &= \frac{K_m}{\tau_m} V''_{\phi} \\
p &= - \frac{1}{I} v_{\phi} + \left(p(0) + \frac{1}{I} v_{\phi}(0) \right) \\
\dot{v}_{\theta} + \frac{1}{\tau_m} v_{\theta} &= \frac{K_m}{\tau_m} V''_{\theta} \\
q &= - \frac{1}{I} v_{\theta} + \left(q(0) + \frac{1}{I} v_{\theta}(0) \right) \\
\dot{v}_{\psi} + \frac{1}{\tau_m} v_{\psi} &= \frac{K_m}{\tau_m} V''_{\psi} \\
r &= - \frac{1}{I} v_{\psi} + \left(r(0) + \frac{1}{I} v_{\psi}(0) \right) .
\end{aligned}
\qquad \text{(A-33)}
$$

Finally the equations for the Euler angles ϕ, θ, ψ are:

$$
\begin{aligned}
\dot{\phi} &= p + (t\theta s\phi)q + (t\theta c\phi)r \\
\dot{\theta} &= (c\phi)q - (s\phi)r \\
\dot{\psi} &= \frac{s\phi}{c\theta} q + \frac{c\phi}{c\theta} r .
\end{aligned}
\qquad \text{(A-34)}
$$

All the equations presented so far are summarized in block diagram form in Fig. A-4 for the sensing and processing devices and in Fig. A-5 for the compensators, actuators, and vehicle. It is clear from Fig. A-4 that for a pair of trackers the tracker rate model contributes four dynamic equations while the angle model contributes only the three Euler equations regardless of the number of trackers. Thus, there is a saving of at least one state variable by using the angle model. Figure A-5 clearly shows that the dynamic dimension of the simplified forward loop model is six whereas the more complicated model is of dimension nine. Thus, by restricting ourselves to the tracker angle model the system dimension is nine for the simplified vehicle model and twelve for the one accounting for gyroscopic torques due to the wheels. Also notice that the simplified model forward loop has a constant input, the initial total angular momentum, and thus is essentially a regulator system.

In order to perform the stability analysis that is planned, the system state equations must take the form

$$\dot{x} = Ax + g(x) \tag{A-35}$$

where x is the state vector of appropriate dimension, A is the matrix of the linear part, and $g(x)$ is the collection of nonlinear terms which have no linear part, i.e.,

$$\lim_{\|x\| \to 0} \frac{\|g(x)\|}{\|x\|} = 0 \tag{A-36}$$

where $\| \ \|$ is the Euclidean norm and in particular $g(0) = 0$. The development here will be restricted to the simplified model (no wheel gyroscopic torques).

We begin by defining a set of variables whose value at equilibrium ($\dot{x} = 0$) will be zero, viz.,

$$\begin{aligned} \phi' &= \phi - \phi_e \\ \theta' &= \theta - \theta_e \\ \psi' &= \psi - \psi_e \end{aligned} \tag{A-37}$$

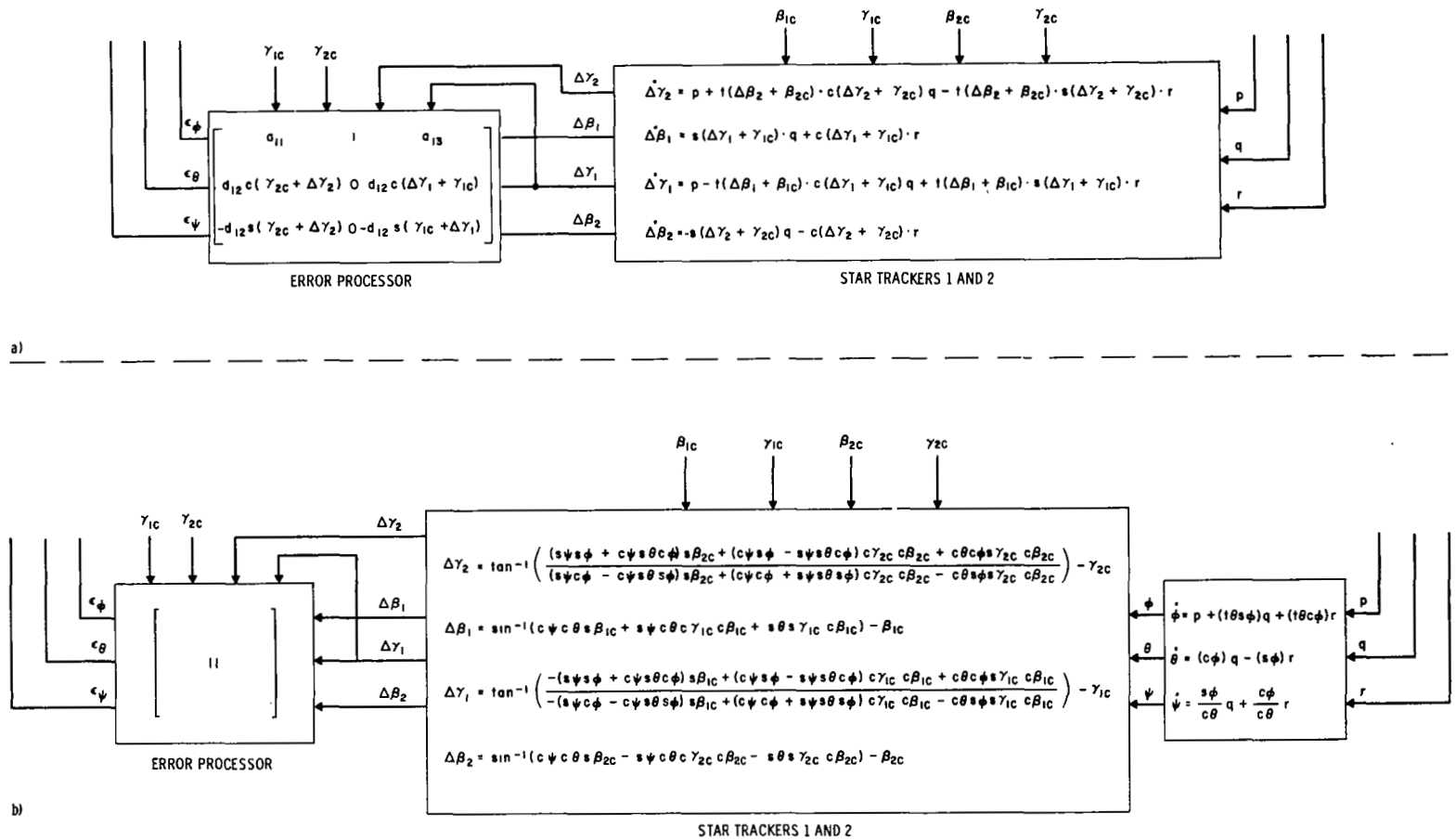


Fig. A-4 Feedback Path. a)Based on Gimbal Angle Rate Equations
b) Based on Euler Angle Rate Equations

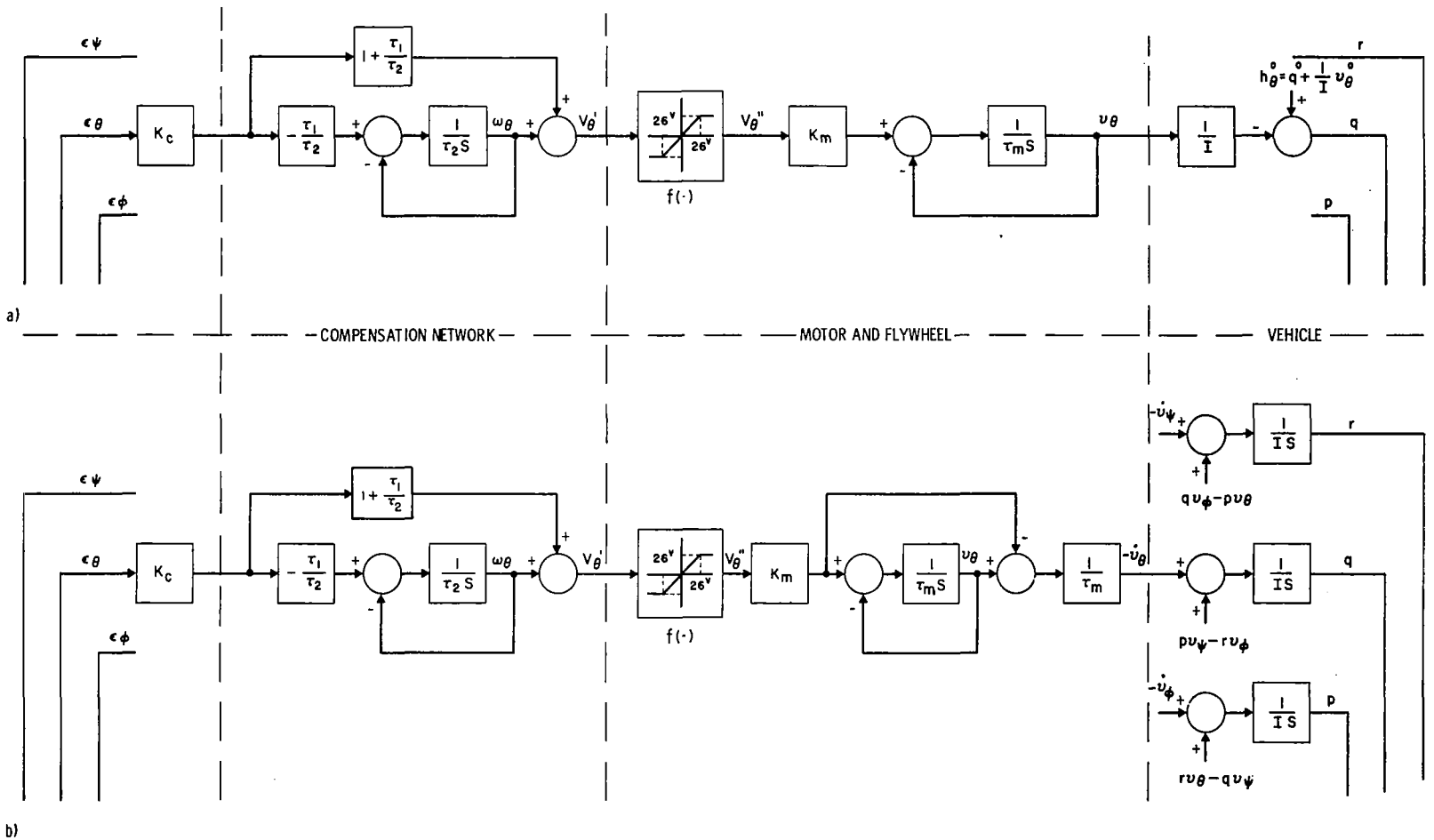


Fig. A-5 Typical Forward Channels. a) Without Wheel Gyroscopic Torques. b) With Wheel Gyroscopic Torques

$$
\begin{aligned}
v'_\phi &= v_\phi - Ih^o_\phi \\
v'_\theta &= v_\theta - Ih^o_\theta \\
v'_\psi &= v_\psi - Ih^o_\psi \\
\omega'_\phi &= \omega_\phi - \left(-\frac{\tau_1 I}{\tau_2 K_m} h^o_\phi\right) \\
\omega'_\theta &= \omega_\theta - \left(-\frac{\tau_1 I}{\tau_2 K_m} h^o_\theta\right) \\
\omega'_\psi &= \omega_\psi - \left(-\frac{\tau_1 I}{\tau_2 K_m} h^o_\psi\right)
\end{aligned}
\qquad \text{(A-37) (Cont.)}
$$

where

$$
\begin{aligned}
h^o_\phi &= p(0) + \frac{1}{I} v_\phi(0) \\
h^o_\theta &= q(0) + \frac{1}{I} v_\theta(0) \\
h^o_\psi &= r(0) + \frac{1}{I} v_\psi(0)
\end{aligned}
\qquad \text{(A-38)}
$$

and ϕ_e, θ_e, ψ_e are the offset angles which are complicated functions of the initial angular momentum Ih^o_ϕ, Ih^o_θ, Ih^o_ψ and the commanded gimbal angles, i.e., they are the values that produce the equilibrium errors:

$$
\begin{aligned}
\epsilon^o_\phi &= \frac{I}{K_c K_m} h^o_\phi \\
\epsilon^o_\theta &= \frac{I}{K_c K_m} h^o_\theta \\
\epsilon^o_\psi &= \frac{I}{K_c K_m} h^o_\psi \,.
\end{aligned}
\qquad \text{(A-39)}
$$

Thus, the equations in the case of the tracker angle model become:

$$\dot{\phi}' = -\frac{1}{I} v'_{\phi} - \frac{(t\theta s\phi)}{I} v'_{\theta} - \frac{(t\theta c\phi)}{I} v'_{\psi}$$

$$\dot{v}'_{\phi} = -\frac{1}{\tau_m} v'_{\phi} + \frac{K_m}{\tau_m} f\left(K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon'_{\phi} + \omega'_{\phi} + \frac{I}{K_m} h^o_{\phi}\right) - \frac{I}{\tau_m} h^o_{\phi}$$

$$\dot{\omega}'_{\phi} = -\frac{1}{\tau_2} \omega'_{\phi} - \frac{K_c \tau_1}{\tau_2^2} \epsilon'_{\phi}$$

$$\dot{\theta}' = -\frac{(c\phi)}{I} v'_{\theta} + \frac{(s\phi)}{I} v'_{\psi}$$

$$\dot{v}'_{\theta} = -\frac{1}{\tau_m} v'_{\theta} + \frac{K_m}{\tau_m} f\left(K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon'_{\psi} + \omega'_{\psi} + \frac{I}{K_m} h^o_{\psi}\right) - \frac{I}{\tau_m} h^o_{\psi} \qquad \text{(A-40)}$$

$$\dot{\omega}'_{\theta} = -\frac{1}{\tau_2} \omega'_{\theta} - \frac{K_c \tau_1}{\tau_2^2} \epsilon'_{\theta}$$

$$\dot{\psi}' = -\frac{s\phi}{Ic\theta} v'_{\theta} - \frac{c\phi}{Ic\theta} v'_{\psi}$$

$$\dot{v}'_{\psi} = -\frac{1}{\tau_m} v'_{\psi} + \frac{K_m}{\tau_m} f\left(K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon'_{\psi} + \omega'_{\psi} + \frac{I}{K_m} h^o_{\psi}\right) - \frac{I}{\tau_m} h^o_{\psi}$$

$$\dot{\omega}'_{\psi} = -\frac{1}{\tau_2} \omega'_{\psi} - \frac{K_c \tau_1}{\tau_2^2} \epsilon'_{\psi}$$

$$\epsilon'_\phi = a_{11}\Delta\beta_1 + \Delta\gamma_1 + a_{13}\Delta\beta_2 - \frac{I}{K_c K_m} h^o_\phi$$

$$\epsilon'_\theta = d_{12}c(\gamma_{2c} + \Delta\gamma_2) \cdot \Delta\beta_1 + d_{12}c(\gamma_{1c} + \Delta\gamma_1) \cdot \Delta\beta_2 - \frac{I}{K_c K_m} h^o_\theta$$

$$\epsilon'_\psi = - d_{12}s(\gamma_{2c} + \Delta\gamma_2) \cdot \Delta\beta_1 - d_{12}s(\gamma_{1c} + \Delta\gamma_1) \cdot \Delta\beta_2 - \frac{I}{K_c K_m} h^o_\psi$$

$$\Delta\beta_1 = \sin^{-1}(c\psi c\theta s\beta_{1c} + s\psi c\theta c\gamma_{1c}c\beta_{1c} + s\theta s\gamma_{1c}c\beta_{1c}) - \beta_{1c}$$

$$\Delta\gamma_1 = \tan^{-1}\frac{1}{D_1}\Big(-(s\psi s\phi + c\psi s\theta c\phi)s\beta_{1c} + (c\psi s\phi - s\psi s\theta c\phi)c\gamma_{1c}c\beta_{1c} + c\theta c\phi s\gamma_{1c}c\beta_{1c}\Big) - \gamma_{1c}$$

$$D_1 = \Big(-(s\psi c\phi - c\psi s\theta s\phi)s\beta_{1c} + (c\psi c\phi + s\psi s\theta s\phi)c\gamma_{1c}c\beta_{1c} - c\theta s\phi s\gamma_{1c}c\beta_{1c}\Big)$$

(A-40) (Cont.)

$$\Delta\beta_2 = \sin^{-1}(c\psi c\theta s\beta_{2c} - s\psi c\theta c\gamma_{2c}c\beta_{2c} - s\theta s\gamma_{2c}c\beta_{2c}) - \beta_{2c}$$

$$\Delta\gamma_2 = \tan^{-1}\frac{1}{D_2}\Big((s\psi s\phi + c\psi s\theta c\phi)s\beta_{2c} + (c\psi s\phi - s\psi s\theta c\phi)c\gamma_{2c}c\beta_{2c} + c\theta c\phi s\gamma_{2c}c\beta_{2c}\Big) - \gamma_{2c}$$

$$D_2 = \Big((s\psi c\phi - c\psi s\theta s\phi)s\beta_{2c} + (c\psi c\phi + s\psi s\theta s\phi)c\gamma_{2c}c\beta_{2c} - c\theta s\phi s\gamma_{2c}c\beta_{2c}\Big) .$$

As stated above in (A-35), we desire to put the system into the form

$$\dot{x} = Ax + g(x) \qquad \text{(A-41)}$$

where A is a constant matrix representing the linear part of the system and

$$g(x) = F(x) - Ax \tag{A-42}$$

where, just as in (A-36),

$$\lim_{\|x\| \to 0} \frac{\|g(x)\|}{\|x\|} = 0 \ . \tag{A-43}$$

A is computed by linearizing $F(x)$ about the equilibrium x_e, i.e.,

$$A = \frac{\partial F(x)}{\partial x}\Big|_{x=x_e} , \tag{A-44}$$

where

$$\frac{\partial F}{\partial x} = \begin{bmatrix} \frac{\partial F_1}{\partial x_1} & \cdots & \frac{\partial F_1}{\partial x_9} \\ \vdots & & \\ \frac{\partial F_9}{\partial x_1} & \cdots & \frac{\partial F_9}{\partial x_9} \end{bmatrix} \tag{A-45}$$

and in this case

$$x_e = 0 \ . \tag{A-46}$$

Performing the above differentiation we obtain Fig. A-6a, representing the linear part of the system, where the A_{ij}'s are given in Fig. A-7.

Since θ_e, ψ_e, and ϕ_e are $\ll 1$ in radians, a Taylor series expansion was made of the A_{ij}'s about $\theta_e = 0$, $\psi_e = 0$, $\phi_e = 0$, retaining the constant terms and the linear terms as good approximations of the values of the A_{ij}'s, i.e.,

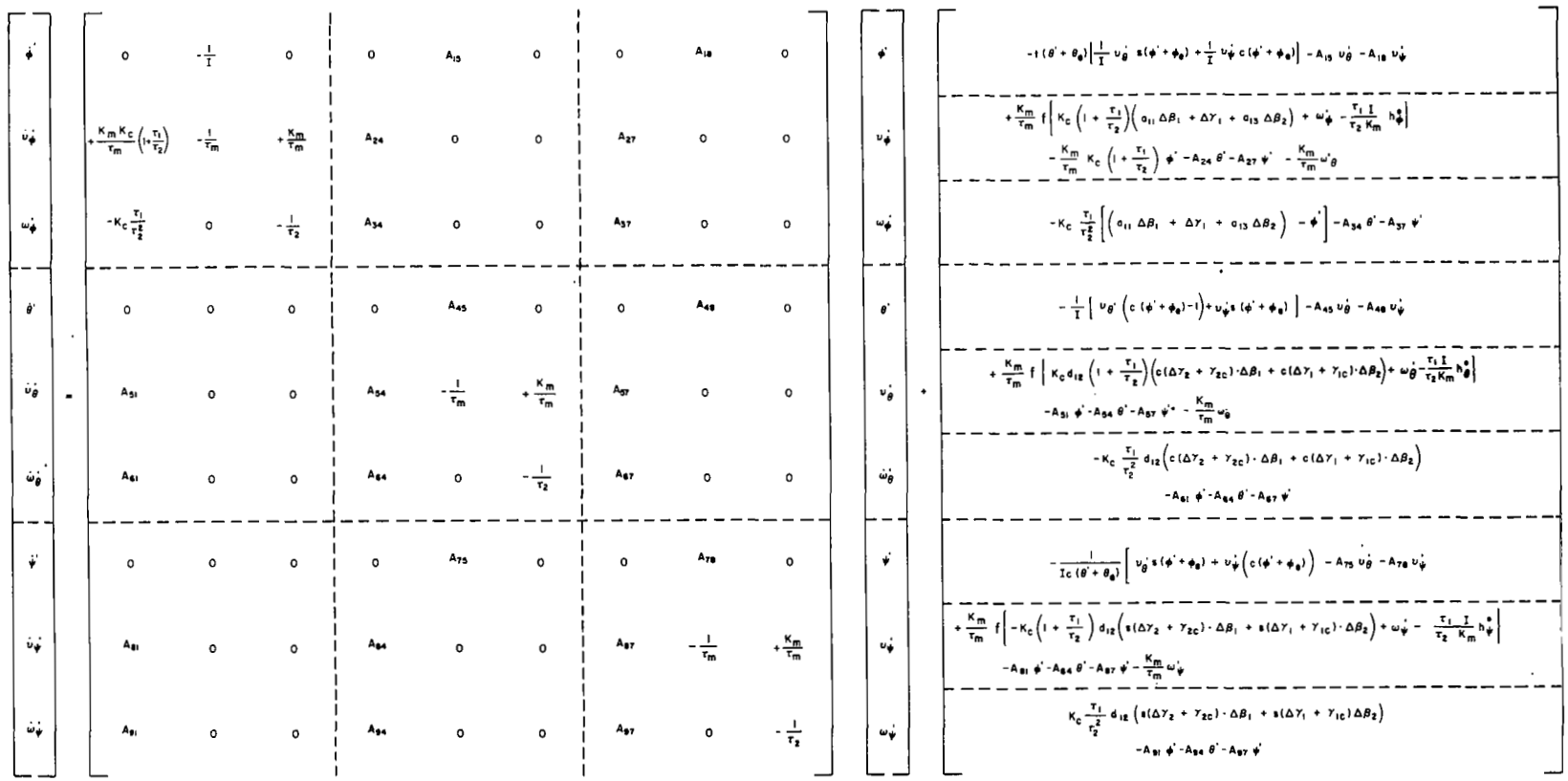


Fig. A-6a State Equations Based on Tracker Angle Model

$$A_{15} = 0$$

$$A_{18} = -\frac{1}{I}\,\phi_e$$

$$A_{24} = -\frac{K_m K_c}{\tau_m}\left(1 + \frac{\tau_1}{\tau_2}\right)a_{24}$$

$$a_{24} = \left\{(a_{11}s\gamma_{1c} - c\gamma_{1c}t\beta_{1c} - a_{13}s\gamma_{2c}) + \left[a_{11}(c\gamma_{1c})^2 t\beta_{1c} - c\gamma_{1c}s\gamma_{1c} - 2(t\beta_{1c})^2 c\gamma_{1c}s\gamma_{1c} - a_{13}t\beta_{2c}(c\gamma_{2c})^2\right]\theta_e + \left[a_{11}s\gamma_{1c}c\gamma_{1c}t\beta_{1c} + \left(\frac{s\gamma_{1c}}{c\beta_{1c}}\right)^2 - 1 + a_{13}t\beta_{2c}s\gamma_{2c}c\gamma_{2c}\right]\psi_e\right\}$$

$$A_{27} = -\frac{K_m K_c}{\tau_m}\left(1 + \frac{\tau_1}{\tau_2}\right)a_{27}$$

$$a_{27} = \left\{(a_{11}c\gamma_{1c} + s\gamma_{1c}t\beta_{1c} - a_{13}c\gamma_{2c}) + \left[a_{11}c\gamma_{1c}s\gamma_{1c}t\beta_{1c} + (s\gamma_{1c})^2\left(1 + 2(t\beta_{1c})^2\right) - \frac{1}{(c\beta_{1c})^2} + a_{13}c\gamma_{2c}s\gamma_{2c}t\beta_{2c}\right]\theta_e + \left[-a_{11}t\beta_{1c}(s\gamma_{1c})^2 + s\gamma_{1c}c\gamma_{1c}\left(1 + 2(t\beta_{1c})^2\right) - a_{13}(s\gamma_{2c})^2 t\beta_{2c}\right]\psi_e\right\}$$

$$A_{34} = -\frac{K_c\tau_1}{\tau_2^2}\,a_{24}$$

$$A_{37} = -\frac{K_c\tau_1}{\tau_2^2}\,a_{27}$$

$$A_{45} = -\frac{1}{I}$$

$$A_{48} = \frac{1}{I}\,\phi_e$$

Fig. A-6b Linearizations of A_{ij} About $\phi_e = \theta_e = \psi_e = 0$ and Linear Estimates of ϕ_e, θ_e, ψ_e

$$A_{51} = -\frac{K_m K_c}{\tau_m}\left(1 + \frac{\tau_1}{\tau_2}\right) d_{12} a_{51}$$

$$a_{51} = -s(\gamma_{1c} - \gamma_{2c})\psi_e$$

$$A_{54} = -\frac{K_m K_c}{\tau_m}\left(1 + \frac{\tau_1}{\tau_2}\right) d_{12} a_{54}$$

$$a_{54} = \left\{s(\gamma_{1c} - \gamma_{2c}) + \left[-t\beta_{1c}c\gamma_{2c}(c\gamma_{1c})^2 - t\beta_{2c}c\gamma_{1c}(c\gamma_{2c})^2 - 2t\beta_{1c}s\gamma_{1c}s\gamma_{2c}c\gamma_{1c} - 2t\beta_{2c}s\gamma_{1c}s\gamma_{2c}c\gamma_{2c}\right]\theta_e + \left[t\beta_{1c}s\gamma_{2c}(s\gamma_{1c})^2 + t\beta_{2c}s\gamma_{1c}(s\gamma_{2c})^2\right]\psi_e\right\}$$

$$A_{57} = -\frac{K_m K_c}{\tau_m}\left(1 + \frac{\tau_1}{\tau_2}\right) d_{12} a_{57}$$

$$a_{57} = \left\{\left[s\gamma_{1c}t\beta_{2c}(s\gamma_{2c})^2 + (s\gamma_{1c})^2 s\gamma_{2c}t\beta_{1c}\right]\theta_e + \left[-(s\gamma_{2c})^2 c\gamma_{1c}t\beta_{2c} + c\gamma_{2c}(s\gamma_{1c})^2 t\beta_{1c}\right]\psi_e + s(\gamma_{1c} - \gamma_{2c})\phi_e\right\}$$

$$A_{61} = -\frac{K_c \tau_1}{\tau_2^2} d_{12} a_{51}$$

$$A_{64} = -\frac{K_c \tau_1}{\tau_2^2} d_{12} a_{54}$$

$$A_{67} = -\frac{K_c \tau_1}{\tau_2^2} d_{12} a_{57}$$

$$A_{75} = \frac{1}{I}\phi_e$$

$$A_{78} = -\frac{1}{I}$$

Fig. A-6b (Cont) Linearizations of A_{ij} About $\phi_e = \theta_e = \psi_e = 0$ and Linear Estimates of ϕ_e, θ_e, ψ_e

$$A_{81} = \frac{K_m K_c}{\tau_m}\left(1 + \frac{\tau_1}{\tau_2}\right) d_{12} a_{81}$$

$$a_{81} = s(\gamma_{1c} - \gamma_{2c}) \cdot \theta_e$$

$$A_{84} = \frac{K_m K_c}{\tau_m}\left(1 + \frac{\tau_1}{\tau_2}\right) d_{12} a_{84}$$

$$a_{84} = \left\{\left[s\gamma_{1c}c_{2c}t\beta_{2c}(1+s\gamma_{2c}) + s\gamma_{2c}c\gamma_{1c}t\beta_{1c}(1+s\gamma_{1c})\right]\theta_e + \left[c\gamma_{1c}c\gamma_{2c}t\beta_{2c}(1+s\gamma_{2c}) + c\gamma_{1c}c\gamma_{2c}t\beta_{1c}(1+s\gamma_{1c})\right]\psi_e + s(\gamma_{1c} - \gamma_{2c})\phi_e\right\}$$

$$A_{87} = \frac{K_m K_c}{\tau_m}\left(1 + \frac{\tau_1}{\tau_2}\right) d_{12} a_{87}$$

$$a_{87} = \left\{-s(\gamma_{1c} - \gamma_{2c}) + \left[-s\gamma_{1c}(c\gamma_{2c})^3 + (c\gamma_{1c})^3 s\gamma_{2c} + t\beta_{1c}c\gamma_{2c}(c\gamma_{1c})^2 + t\beta_{2c}c\gamma_{1c}(c\gamma_{2c})^2\right]\theta_e \right.$$
$$\left. + \left[-c\gamma_{1c}(c\gamma_{2c})^3 + (c\gamma_{1c})^3 c\gamma_{2c} - t\beta_{2c}s\gamma_{1c}(s\gamma_{2c})^2 - t\beta_{1c}s\gamma_{2c}(s\gamma_{1c})^2 - 2c\gamma_{1c}s\gamma_{2c}c\gamma_{2c}t\beta_{2c} - 2s\gamma_{1c}c\gamma_{2c}c\gamma_{1c}t\beta_{1c}\right]\psi_e\right\}$$

$$A_{91} = \frac{K_c \tau_1}{\tau_2^2} d_{12} a_{81}$$

$$A_{94} = \frac{K_c \tau_1}{\tau_2^2} d_{12} a_{84}$$

$$A_{97} = \frac{K_c \tau_1}{\tau_2^2} d_{12} a_{87}$$

Fig. A-6b (Cont) Linearizations of A_{ij} About $\phi_e = \theta_e = \psi_e = 0$ and Linear Estimates of ϕ_e, θ_e, ψ_e

$$\theta_e = \frac{I}{K_m K_c} \frac{1}{d_{12} s(\gamma_{1c} - \gamma_{2c})} h_\theta^o$$

$$\psi_e = -\theta \frac{I}{K_m K_c} \frac{1}{d_{12} s(\gamma_{1c} - \gamma_{2c})} h_\psi^o$$

$$\phi_e = \frac{I}{K_m K_c} \left\{ h_\phi^o - \frac{(a_{11} s\gamma_{1c} - a_{13} s\gamma_{2c} - c\gamma_{1c} t\beta_{1c})}{d_{12} s(\gamma_{1c} - \gamma_{2c})} h_\theta^o + \frac{(a_{11} c\gamma_{1c} - a_{13} c\gamma_{2c} + s\gamma_{1c} t\beta_{1c})}{d_{12} s(\gamma_{1c} - \gamma_{2c})} h_\psi^o \right\}$$

Fig. A-6b (Cont) Linearizations of A_{ij} About $\phi_e = \theta_e = \psi_e = 0$ and Linear Estimates of ϕ_e, θ_e, ψ_e

$$A_{15} = \frac{-t\theta_e s\beta_e}{I}$$

$$A_{18} = \frac{-t\theta_e c\beta_e}{I}$$

$$\nu_1 = -(s\theta_e c\psi_e) s\beta_{1c} - (s\theta_e s\psi_e) c\beta_{1c} c\gamma_{1c} + c\theta_e s\gamma_{1c} c\beta_{1c}$$

$$\iota_1 = \Big[(1 - c^2\psi_e c^2\theta_e) s^2\beta_{1c} - 2(s\psi_e c\psi_e c^2\theta_e) s\beta_{1c} c\gamma_{1c} c\beta_{1c} - 2(c\psi_e s\theta_e c\theta_e) s\beta_{1c} s\gamma_{1c} c\beta_{1c} + (1 - s^2\psi_e c^2\theta_e) c^2\gamma_{1c} c^2\beta_{1c}$$
$$- 2(c\theta_e s\theta_e s\psi_e) c\gamma_{1c} c^2\beta_{1c} s\gamma_{1c} + c^2\theta_e s^2\gamma_{1c} c^2\beta_{1c}\Big]^{1/2}$$

$$\nu_2 = (s\psi_e c\psi_e c\theta_e) s^2\beta_{1c} + c\theta_e (s^2\psi_e - c^2\psi_e) s\beta_{1c} c\beta_{1c} c\gamma_{1c} + (s\psi_e s\theta_e) s\beta_{1c} c\beta_{1c} s\gamma_{1c} - (c\psi_e s\psi_e c\theta_e) c^2\gamma_{1c} c^2\beta_{1c}$$
$$- (s\theta_e c\theta_e) c^2\beta_{1c} c\gamma_{1c} s\gamma_{1c}$$

$$\iota_2 = \iota_1^2$$

$$\nu_3 = -(s\theta_e c\psi_e) s\beta_{2c} + (s\theta_e s\psi_e) c\beta_{2c} c\gamma_{2c} - c\theta_e c\beta_{2c} s\gamma_{2c}$$

$$\iota_3 = \Big[(1 - c^2\psi_e c^2\theta_e) s^2\beta_{2c} + 2(s\psi_e c\psi_e c^2\theta_e) s\beta_{2c} c\gamma_{2c} c\beta_{2c} + 2(c\psi_e s\theta_e c\theta_e) s\beta_{2c} s\gamma_{2c} c\beta_{2c} + (1 - s^2\psi_e c^2\theta_e) c^2\gamma_{2c} c^2\beta_{2c}$$
$$- 2(c\theta_e s\theta_e s\psi_e) c\gamma_{2c} c^2\beta_{2c} s\gamma_{2c} + c^2\theta_e s^2\gamma_{2c} c^2\beta_{2c}\Big]$$

$$A_{24} = \frac{-K_m K_c}{\tau_m}(1 + \tau_1/\tau_2)\left[a_{11}\frac{\nu_1}{\iota_1} + \frac{\nu_2}{\iota_2} + a_{13}\frac{\nu_3}{\iota_3}\right]$$

$$A_{34} = \frac{-K_c \tau_1}{\tau_2^2}\left[a_{11}\frac{\nu_1}{\iota_1} + \frac{\nu_2}{\iota_2} + a_{13}\frac{\nu_3}{\iota_3}\right]$$

$$\nu_4 = -(s\psi_e c\theta_e) s\beta_{1c} + (c\psi_e c\theta_e) c\psi_{1c} c\beta_{1c}$$

$$\iota_4 = \Big[(1 - c^2\psi_e c^2\theta_e) s^2\beta_{1c} - 2(s\psi_e c\psi_e c^2\theta_e) s\beta_{1c} c\gamma_{1c} c\beta_{1c} - 2(c\psi_e s\theta_e c\theta_e) s\beta_{1c} s\gamma_{1c} c\beta_{1c} + (1 - s^2\psi_e c^2\theta_e) c^2\gamma_{1c} c^2\beta_{1c}$$
$$- 2(c\theta_e s\theta_e s\psi_e) c\gamma_{1c} c^2\beta_{1c} s\gamma_{1c} + c^2\theta_e s^2\gamma_{1c} c^2\beta_{1c}\Big]^{1/2}$$

Fig. A-7 The Elements of Matrix A

$$\nu_5 = -s\theta_e s^2\beta_{1c} + (c\theta_e c\psi_e) s\beta_{1c} s\gamma_{1c} c\beta_{1c} - s\theta_e c^2\gamma_{1c} c^2\beta_{1c} + c\theta_e s\psi_e s\gamma_{1c} c^2\beta_{1c} c\gamma_{1c}$$

$$\iota_5 = \iota_4^2$$

$$\nu_6 = -(s\psi_e c\theta_e) s\beta_{2c} - (c\psi_e c\theta_e) c\beta_{2c} c\gamma_{2c}$$

$$\iota_6 = \Big[(1 - c^2\psi_e c^2\theta_e) s^2\beta_{2c} + 2 (s\psi_e c\psi_e c^2\theta_e) s\beta_{2c} c\gamma_{2c} c\beta_{2c} + 2 (c\psi_e s\theta_e c\theta_e) s\beta_{2c} s\gamma_{2c} c\beta_{2c} + (1 - s^2\psi_e c^2\theta_e) c^2\gamma_{2c} c^2\beta_{2c}$$
$$- 2 (c\theta_e s\theta_e s\psi_e) c\gamma_{2c} c^2\beta_{2c} s\gamma_{2c} + c^2\theta_e s^2\gamma_{2c} c^2\beta_{2c} \Big]^{1/2}$$

$$A_{27} = \frac{-K_m K_c}{\tau_m}(1 + \tau_1/\tau_2) \left[a_{11} \frac{\nu_4}{\iota_4} + \frac{\nu_5}{\iota_5} + a_{13} \frac{\nu_6}{\iota_6} \right]$$

$$A_{37} = \frac{K_c \tau_1}{\tau_2^2} \left[a_{11} \frac{\nu_4}{\iota_4} + \frac{\nu_5}{\iota_5} + a_{13} \frac{\nu_6}{\iota_6} \right]$$

$$A_{45} = \frac{-c\beta_e}{I}$$

$$A_{48} = \frac{s\beta_e}{I}$$

$$\nu_7 = \left[(s\psi_e s\beta_e + c\psi_e c\beta_e s\theta_e) s\beta_{2c} + (c\psi_e s\beta_e - s\psi_e s\theta_e c\beta_e) c\gamma_{2c} c\beta_{2c} + (c\theta_e c\beta_e) s\gamma_{2c} c\beta_{2c} \right] \left[s^{-1}(c\psi_e c\theta_e s\beta_{1c} + (s\psi_e c\theta_e) c\gamma_{1c} c\beta_{1c} + s\theta_e s\gamma_{1c} c\beta_{1c}) - \beta_{1c} \right]$$

$$\iota_7 = \left[(1 - c^2\psi_e c^2\theta_e) s^2\beta_{2c} + 2 (s\psi_e c\psi_e c^2\theta_e) s\beta_{2c} c\gamma_{2c} c\beta_{2c} + 2 (c\psi_e c\theta_e s\theta_e) s\beta_{2c} s\gamma_{2c} c\beta_{2c} + (1 - s^2\psi_e c^2\theta_e) c^2\beta_{2c} c^2\gamma_{2c} - 2 (c\theta_e s\theta_e s\psi_e) c^2\beta_{2c} c\gamma_{2c} s\gamma_{2c} + c^2\theta_e s^2\gamma_{2c} c^2\beta_{1c} \right]^{1/2}$$

$$\nu_8 = \left[-(s\psi_e s\beta_e + c\psi_e c\theta_e s\theta_e) s\beta_{1c} + (c\psi_e s\beta_e - s\psi_e s\theta_e c\beta_e) c\gamma_{1c} c\beta_{1c} + (c\theta_e c\beta_e) s\gamma_{1c} c\beta_{1c} \right] \left[s^{-1} \left((c\psi_e c\theta_e) s\beta_{2c} - (s\psi_e c\theta_e) c\gamma_{2c} c\beta_{2c} - s\theta_e s\gamma_{2c} c\beta_{2c} \right) - \beta_{2c} \right]$$

$$\iota_8 = \left[(1 - c^2\psi_e c^2\theta_e) s^2\beta_{1c} - 2 (s\psi_e c\psi_e c^2\theta_e) s\beta_{1c} c\gamma_{1c} c\beta_{1c} - 2 (c\psi_e c\theta_e s\theta_e) s\beta_{1c} s\gamma_{1c} c\beta_{1c} + (1 - s^2\psi_e c^2\theta_e) c^2\beta_{1c} c^2\gamma_{1c} - 2 (c\theta_e s\theta_e s\psi_e) c^2\beta_{1c} c\gamma_{1c} s\gamma_{1c} + c^2\theta_e s^2\gamma_{2c} c^2\beta_{12} \right]^{1/2}$$

Fig. A-7 (Cont) The Elements of Matrix A

$$A_{51} = \frac{-K_m K_c (1 + \tau_1/\tau_2)\, d_{12}}{\tau_m} \left(\frac{\nu_7}{\delta_7} + \frac{\nu_8}{\delta_8} \right)$$

$$\nu_9 = (c\psi_e\, c\theta_e\, s^2\psi_e\, s\theta_e\, s\beta_e + s\psi_e\, s\theta_e\, c^2\psi_e\, c\theta_e\, c\beta_e)\, s^3\beta_{2c} + (s\psi_e\, s\beta_e\, c^2\psi_e\, c\theta_e - s^2\psi_e s\theta_e\, c\psi_e\, c\theta_e\, c\beta_e)\, s^2\beta_{2c}\, c\gamma_{2c}\, c\beta_{2c} + (s\psi_e\, c\psi_e\, c^2\theta_e\, c\beta_e)\, s^2\beta_{2c}\, s\gamma_{2c}\, c\beta_{2c}$$

$$- (s^3\psi_e\, s\beta_e + s^2\psi_e\, s\theta_e\, c\psi_e\, c\beta_e - s\psi_e\, s\beta_e\, c^2\psi_e - c^3\psi_e\, c\beta_e\, s\theta_e)\, (c\theta_e)\, s^2\beta_{2c}\, c\gamma_{2c}\, c\beta_{2c} - c\theta_e\, (s^2\psi_e\, s\beta_e\, c\psi_e - s^3\psi_e\, s\theta_e\, c\beta_e - c^3\psi_e\, s\beta_e + s\psi_e\, s\theta_e\, c^2\psi_e c\beta_e)\, c^2\gamma_{2c}\, c^2\beta_{2c}\, s\beta_{2c}$$

$$- c^2\theta_e\, (s^2\psi_e - c^2\psi_e)\, s\gamma_{2c}\, s\beta_{2c}\, c^2\beta_{2c} c\gamma_{2c} - (s^2\psi_e\, s\beta_e\, c\psi_e\, c\theta_e + s\psi_e\, s\theta_e\, c^2\psi_e\, c\theta_e)\, c^2\gamma_{2c}\, c^2\beta_{2c}\, s\beta_{2c} - (s\psi_e\, s\beta_e\, c^2\psi_e\, c\theta_e - s^2\psi_e\, s\beta_e\, c\psi_e\, c\theta_e\, c\beta_e)\, c^3\gamma_{2c}\, c^3\beta_{2c}$$

$$- (s\psi_e\, c\psi_e\, c^2\theta_e\, c\beta_e)\, s\gamma_{2c}\, c^3\beta_{2c}\, c\gamma_c - (s^2\psi_e\, s\theta_e\, s\beta_e + s\psi_e\, s^2\theta_e\, c\psi_e\, c\beta_e)\, s^2\beta_{2c}\, s\gamma_{2c}\, c\beta_{2c} - (s\psi_e\, s\theta_e\, c\psi_e\, c\beta_e - s^2\psi_e\, s^2\theta_e\, c\beta_e)\, c\gamma_{2c}\, c^2\beta_{2c}\, s\gamma_{2c}\, s\beta_{2c}$$

$$- (s\psi_e\, s\theta_e\, c\theta_e\, c\beta_e)\, s^2\gamma_{2c}\, s\beta_{2c}\, c^2\beta_{2c} - (s\psi_e\, s\theta_e\, s\beta_e\, c\psi_e + c^2\psi_e\, c\beta_e\, s^2\theta_e)\, s\gamma_{2c}\, s\beta_{2c}\, c\gamma_{2c}\, c^2\beta_{2c} - (c^2\psi_e\, s\theta_e\, s\beta_e - s\psi_e\, s^2\theta_e\, c\psi_e\, c\beta_e)\, c^2\gamma_{2c}\, c^3\beta_{2c}\, s\gamma_{2c}$$

$$- (c\psi_e\, c\theta_e\, c\beta_e\, s\theta_e)\, s^2\gamma_{2c}\, c^3\beta_{2c}\, c\gamma_{2c}$$

$$\delta_9 = \left[\left((s\psi_e\, s\beta_e + c\psi_e\, s\theta_e\, c\beta_e)\, s\beta_{2c} + (c\psi_e\, s\beta_e - s\psi_e\, s\theta_e\, c\beta_e)\, c\gamma_{2c}\, c\beta_{2c} + c\theta_e\, c\beta_e\, s\gamma_{2c}\, c\beta_{2c} \right)^2 + \left((s\psi_e\, c\beta_e - c\psi_e\, s\theta_e\, s\beta_e)\, 2\beta_{2c} + (c\psi_e\, c\beta_e + s\psi_e\, s\theta_e\, s\beta_e)\, c\gamma_{2c}\, c\beta_{2c} - c\theta_e\, s\beta_e\, s\gamma_{2c}\, c\beta_{2c} \right)^2 \right]^{3/2}$$

$$\nu_{10} = -(c\psi_e\, c\theta_e\, s^2\psi_e\, s\theta_e\, s\beta_e + s\psi_e\, s\theta_e c^2\psi_e\, c\theta_e c\beta_e)\, s^3\beta_{1c} + (s\psi_e\, s\beta_e\, c^2\psi_e\, c\theta_e - s^2\psi_e\, s\theta_e\, c\psi_e\, c\theta_e\, c\beta_e)\, s^2\beta_{1c}\, s\gamma_{1c}\, c\beta_{1c} - (s\psi_e\, c\psi_e\, c^2\theta_e\, c\beta_e)\, s^2\beta_{1c}\, s\gamma_{1c}\, c\beta_{1c}$$

$$- (s^3\psi_e\, s\beta_e + s^2\psi_e\, s\theta_e\, c\psi_e\, c\beta_e - s\psi_e\, s\beta_e c^2\psi_e - c^3\psi_e c\beta_e\, s\theta_e)\, (c\theta_e)\, s^2\beta_{1c}\, c\gamma_{1c}\, c\beta_{1c} + c\theta_e\, (s^2\psi_e\, s\beta_e\, c\psi_e - s^3\psi_e\, s\theta_e\, c\beta_e - c^3\psi_e\, s\beta_e + s\psi_e\, s\theta_e\, c^2\psi_e\, c\beta_e)\, c^2\gamma_{1c}\, c^2\beta_{1c}\, s\beta_{1c}$$

$$+ (c^2\theta_e)\, (s^2\psi_e - c^2\psi_e)\, s\gamma_{1c}\, s\beta_{1c}\, c^2\beta_{1c}\, c\gamma_{1c} + (s^2\psi_e\, s\beta_e\, c\psi_e\, c\theta_e + s\psi_e\, s\theta_e\, c^2\psi_e\, c\theta_e)\, c^2\gamma_{1c}\, c^2\beta_{1c}\, s\beta_{1c} - (s\psi_e\, s\beta_e\, c^2\psi_e\, c\theta_e - s^2\psi_e\, s\beta_e\, c\psi_e\, c\theta_e\, c\beta_e)\, c^3\gamma_{1c}\, c^3\beta_{2c}$$

$$- (s\psi_e\, c\psi_e\, c^2\theta_e\, c\beta_e)\, s\gamma_{1c}\, c^3\beta_{1c}\, c\gamma_{1c} - (s^2\psi_e\, s\theta_e\, s\beta_e + s\psi_e\, s^2\theta_e\, c\psi_e\, c\beta_e)\, s^2\beta_{1c}\, s\gamma_{1c}\, c\beta_{1c} + (s\psi_e\, s\theta_e\, c\psi_e\, c\beta_e - s^2\psi_e s^2\theta_e\, c\beta_e)\, c\gamma_{1c}\, c^2\beta_{1c}\, s\gamma_{1c}\, s\beta_{1c}$$

$$+ (s\psi_e\, s\theta_e\, c\theta_e\, c\beta_e)\, s\gamma_{1c}\, s\beta_{1c}\, c^2\beta_{1c} + (s\psi_e\, s\theta_e\, s\beta_e\, c\psi_e + c^2\psi_e\, c\beta_e s^2\theta_e)\, s\gamma_{1c}\, s\beta_{1c}\, c\gamma_{1c}\, c^2\beta_{1c} - (c^2\psi_e\, s\theta_e\, s\beta_e - s\psi_e\, s^2\theta_e\, c\psi_e\, c\beta_e)\, c^2\gamma_{1c}\, c^3\beta_{1c}\, s\gamma_{1c}$$

$$- (c\psi_e\, c\theta_e\, c\beta_e\, s\theta_e)\, s^2\gamma_{1c}\, c^3\beta_{1c}\, c\gamma_{1c}$$

$$\delta_{10} = \left\{ \left[-(s\psi_e s\beta_e + c\psi_e\, s\theta_e\, c\beta_e)\, s\beta_{1c} + (c\psi_e\, s\beta_e - s\psi_e\, s\theta_e\, c\beta_e)\, c\gamma_{1c}\, c\beta_{1c} + c\theta_e\, c\beta_e\, s\gamma_{1c}\, c\beta_{1c} \right]^2 + \left[-(s\psi_e c\beta_e - c\psi_e\, s\theta_e\, s\beta_e)\, s\beta_{1c} + (c\psi_e\, c\beta_e + s\psi_e\, s\theta_e s\beta_e)\, c\gamma_{1c}\, c\beta_{1c} - c\theta_e\, s\beta_e\, s\gamma_{1c}\, c\beta_{1c} \right]^2 \right\}^{3/2}$$

$$\nu_{11} = \left[-c\beta_e\, s\theta_e\, s\beta_{2c}\, c\gamma_{1c}\, c\beta_{1c} - c\beta_e\, s\theta_e\, c\gamma_{2c}\, c\beta_{2c}\, s\beta_{1c} + s\beta_e\, c\beta_e\, c\theta_e\, s\beta_{2c}\, c\beta_{1c}\, s\gamma_{1c} + s\psi_e\, c\beta_e\, c\theta_e\, s\beta_{1c}\, s\gamma_{2c}\, c\beta_{2c} + c\psi_e\, c\beta_e\, c\theta_e\, c\gamma_{2c}\, c\beta_{2c}\, c\beta_{1c}\, s\gamma_{1c} \right.$$

$$\left. - c\psi_e c\beta_e\, c\theta_e\, c\beta_{1c}\, c\gamma_{1c}\, c\beta_{2c}\, s\gamma_2 \right]$$

$$\delta_{11} = (\delta_{10})^{1/3}\, (\delta_9)^{1/3}$$

Fig. A-7 (Cont) The Elements of Matrix A

$$A_{54} = \frac{-K_m K_c (1 + \tau_1/\tau_2)}{\tau_m} d_{12} \left(- \Delta\beta_1^e \frac{\nu_9}{\delta_9} - \Delta\beta_2^e \frac{\nu_{10}}{\delta_{10}} + \frac{\nu_{11}}{\delta_{11}} \right)$$

$$\nu_{12} = \Big[-(s\psi_e s\theta_e s\phi_e + c\psi_e c\phi_e s^2\theta_e) s^3\beta_{2c} - (c\psi_e s\theta_e s\phi_e - s\psi_e s^2\theta_e c\phi_e) s^2\beta_{2c} c\gamma_{2c} c\beta_{2c} - (s\theta_e c\theta_e c\phi_e) s^2\beta_{2c} s\gamma_{2c} c\beta_{2c} - (s\psi_e s\theta_e s\phi_e + c\psi_e c\phi_e s^2\theta_e) s\beta_{2c} c^2\gamma_{2c} c^2\beta_{2c}$$
$$- (c\psi_e s\theta_e s\phi_e - s\psi_e s^2\theta_e c\phi_e) c^3\gamma_{2c} c^3\beta_{2c} - (s\theta_e c\theta_e c\phi_e) s\gamma_{2c} c^2\gamma_{2c} c^3\beta_{2c} - (s\psi_e s\phi_e c\psi_e c\theta_e + c^2\psi_e c\theta_e c\phi_e s\theta_e) s^2\beta_{2c} s\gamma_{2c} c\beta_{2c} - (c^2\psi_e c\theta_e s\phi_e - s\psi_e s\theta_e c\psi_e c\theta_e c\phi_e) c\gamma_{2c} c^2\beta_{2c} s\gamma_{2c} s\beta_{2c}$$
$$- (c\psi_e c^2\theta_e c\phi_e) s^2\gamma_{2c} s\beta_{2c} + (s^2\psi_e s\phi_e c\theta_e + s\psi_e s\theta_e c\psi_e c\theta_e c\phi_e) s\gamma_{2c} s\beta_{2c} c^2\beta_{2c} c\gamma_{2c} + (s\psi_e s\phi_e c\psi_e c\theta_e - s^2\psi_e s\theta_e c\theta_e c\phi_e) s\gamma_{2c} c^3\beta_{2c} c^2\gamma_{2c} + (s\psi_e c^2\theta_e c\phi_e) s^2\gamma_{2c} c^3\beta_{2c} c\gamma_{2c} \Big]$$

$$\delta_{12} = \delta_9$$

$$\nu_{13} = (s\psi_e s\theta_e s\phi_e + c\psi_e c\phi_e s^2\theta_e) s^3\beta_{1c} - (c\psi_e s\theta_e s\phi_e - s\psi_e s^2\theta_e c\phi_e) s^2\beta_{1c} c\gamma_{1c} c\beta_{1c} - (s\theta_e c\theta_e c\phi_e) s^2\beta_{1c} s\gamma_{1c} c\beta_{1c} + (s\psi_e s\theta_e s\phi_e + c\psi_e c\phi_e s^2\theta_e) s\beta_{1c} c^2\gamma_{1c} c^2\beta_{1c}$$
$$- (c\psi_e s\theta_e s\theta_e - s\psi_e s^2\theta_e c\phi_e) c^3\gamma_{1c} c^3\beta_{1c} - (s\theta_e c\theta_e c\phi_e) s\gamma_{1c} c^2\gamma_{1c} c^3\beta_{1c} - (s\psi_e s\phi_e c\psi_e c\theta_e + c^2\psi_e c\theta_e c\phi_e s\theta_e) s^2\beta_{1c} s\gamma_{1c} c\beta_{1c} + (c^2\psi_e c\theta_e s\phi_e - s\psi_e s\theta_e c\psi_e c\theta_e c\phi_e)$$
$$c\gamma_{1c} c^2\beta_{1c} s\gamma_{1c} s\beta_{1c} + (c\psi_e c^2\theta_e c\phi_e) s^2\gamma_{1c} s\beta_{1c} c^2\beta_{1c} - (s^2\psi_e s\phi_e c\theta_e + s\psi_e s\theta_e c\psi_e c\theta_e c\phi_e) s\gamma_{1c} s\beta_{1c} c^2\beta_{1c} c\gamma_{1c} + (s\psi_e s\phi_e c\psi_e c\theta_e - s^2\psi_e s\theta_e c\theta_e c\phi_e)$$
$$s\gamma_{1c} c^3\beta_{1c} c^2\gamma_{1c} + (s\psi_e c^2\theta_e c\phi_e) s^2\gamma_{1c} c^3\beta_{1c} c\gamma_{1c}$$

$$\delta_{13} = \delta_{10}$$

$$\nu_{14} = -(s\theta_e s\phi_e c\phi_e) s\beta_{2c} c\gamma_{1c} c\beta_{1c} - (s\theta_e s\phi_e c\phi_e) s\beta_{1c} c\gamma_{2c} c\beta_{1c} + (s\psi_e s\phi_e c^2\theta_e) s\beta_{2c} s\gamma_{1c} c\beta_{1c} + (s\psi_e s\phi_e c^2\theta_e) s\beta_{1c} s\gamma_{2c} c\beta_{2c} + (c\psi_e c^2\theta_e s\phi_e) c\gamma_{2c} c\beta_{2c} s\gamma_{1c} c\beta_{1c}$$
$$- (c\psi_e c^2\theta_3 s\phi_e) c\gamma_{1c} c\beta_{1c} s\gamma_{2c} c\beta_{2c})$$

$$\delta_{14} = \delta_{11}$$

$$A_{57} = \frac{-K_m K_c}{\tau_m} (1 + \tau_1/\tau_2) d_{12} \left(-\Delta\beta_1^e \frac{\nu_{12}}{\delta_{12}} - \Delta\beta_2^e \frac{\nu_{13}}{\delta_{13}} + \frac{\nu_{14}}{\delta_{14}} \right)$$

Fig. A-7 (Cont) The Elements of Matrix A

$$A_{61} = \frac{-K_c \tau_1}{\tau_2^2} d_{12} \left(\frac{\nu_7}{\delta_7} + \frac{\nu_8}{\delta_8} \right)$$

$$A_{64} = \frac{-K_c \tau_1}{\tau_2^2} d_{12} \left(- \Delta\beta_1^e \frac{\nu_9}{\delta_9} - \Delta\beta_2^e \frac{\nu_{10}}{\delta_{10}} + \frac{\nu_{11}}{\delta_{11}} \right)$$

$$A_{67} = \frac{-K_c \tau_1}{\tau_2^2} d_{12} \left(- \Delta\beta_1^e \frac{\nu_{12}}{\delta_{12}} - \Delta\beta_2^e \frac{\nu_{13}}{\delta_{13}} + \frac{\nu_{14}}{\delta_{14}} \right)$$

$$\nu_{15} = (s\psi_e c\beta_e - s\theta_e s\beta_e c\psi_e) s\beta_{2c} + (c\psi_e c\beta_e + s\psi_e s\theta_e s\beta_e) c\gamma_{2c} c\beta_{2c} - (s\psi_e c\theta_e) s\gamma_{2c} c\beta_{2c}$$

$$\delta_{15} = (\delta_9)^{1/3}$$

$$\nu_{16} = -(s\psi_e c\beta_e - s\theta_e s\beta_e c\psi_e) s\beta_{1c} + (c\psi_e c\beta_e + s\psi_e s\theta_e s\beta_e) c\gamma_{1c} c\beta_{1c} - (s\psi_e c\theta_e) s\gamma_{1c} c\beta_{1c}$$

$$\delta_{16} = (\delta_{10})^{1/3}$$

$$A_{81} = \frac{-K_m K_c}{\tau m} (1 + \tau_1/\tau_2)(-d_{12}) \left(\Delta\beta_1^e \frac{\nu_{15}}{\delta_{15}} + \Delta\beta_2^e \frac{\nu_{16}}{\delta_{16}} \right)$$

$$\begin{aligned}
\nu_{17} = {} & (s^2\psi_e c\psi_e c\theta_e c\phi_e - s\psi_e s\theta_e s\phi_e c^2\phi_{2c} c\theta_e) s^3\beta_{2c} + (s\psi_e c^2\psi_e c\theta_e c\phi_e + s^2\psi_e s\theta_e s\phi_e c\psi_e c\theta_e) s^2\beta_{2c} c\gamma_{2c} c\beta_{2c} - (s\psi_e s\phi_e c\psi_e c^2\theta_e) s^2\beta_{2c} s\gamma_{2c} c\beta_{2c} \\
& - (c\theta_e)(s^3\psi_e c\phi_e - s^2\psi_e s\phi_e s\theta_e c\psi_e - s\psi_e c^2\psi_e c\phi_e + c^3\psi_e s\theta_e s\phi_e) s^2\beta_{2c} c\gamma_{2c} c\beta_{2c} - (c\theta_e)(s^2\psi_e c\psi_e c\phi_e + s^3\psi_e s\theta_e s\phi_e - c^3\psi_e c\phi_e - s\psi_e s\theta_e s\phi_e c^2\psi_e) c\gamma_{2c} c^2\beta_{2c} s\beta_{2c} \\
& + (c^2\theta_e s\phi_e)(s^2\psi_e - c^2\psi_e) s\gamma_{2c} s\beta_{2c} c^2\beta_{2c} - (s^2\psi_e c\psi_e c\theta_e c\phi_e - s\psi_e s\theta_e s\phi_e c^2\psi_e c\theta_e) s\beta_{2c} c^2\gamma_{2c} c^2\beta_{2c} - (s\psi_e c^2\psi_e c\theta_e c\phi_e + s^2\psi_e s\theta_e s\phi_e c\psi_e c\theta_e) c^3\gamma_{2c} c^3\beta_{2c} \\
& + (s\psi_e s\phi_e c\psi_e c^2\theta_e) s\gamma_{2c} c^3\beta_{2c} c^2\gamma_{2c} - (s^2\psi_e s\theta_e c\phi_e - s\psi_e s^2\theta_e s\phi_e c\psi_e) s^2\beta_{2c} s\gamma_{2c} c\beta_{2c} - (s\psi_e s\theta_e c\psi_e c\phi_e + s^2\psi_e s^2\theta_e s\phi_e) c\gamma_{2c} c^2\beta_{2c} s\beta_{2c} s\gamma_{2c} \\
& + (s\psi_e s\theta_e s\phi_e c\theta_e) s^2\gamma_{2c} s\beta_{2c} c^2\beta_{2c} - (s\psi_e s\theta_e c\psi_e c\phi_e - c^2\psi_e c\theta_e c\phi_e) s\beta_{2c} s\gamma_{2c} c\gamma_{2c} c^2\beta_{2c} - (c^2\psi_e c\phi_e s\theta_e + s\psi_e s^2\theta_e s\phi_e c\psi_e) c^2\gamma_{2c} c^3\beta_{2c} s\gamma_{2c} \\
& + (c\psi_e c\theta_e s\theta_e s\phi_e) s^2\gamma_{2c} c^3\beta_{2c} c\gamma_{2c}
\end{aligned}$$

$$\delta_{17} = \delta_9$$

Fig. A-7 (Cont) The Elements of Matrix A

$$\nu_{18} = -(s^2\psi_e c\psi_e c\theta_e c\phi_e - s\psi_e s\theta_e s\phi_e c^2\phi_e c\theta_e)s^3\beta_{1c} + (s\psi_e c^2\psi_e c\theta_e c\phi_e + s^2\psi_e s\theta_e s\phi_e c\psi_e c\theta_e)s^2\beta_{1c}c\gamma_{1c}c\beta_{1c} - (s\psi_e s\phi_e c\psi_e c^2\theta_e)s^2\beta_{1c}s\gamma_{1c}c\beta_{1c}$$

$$- (c\theta_e)(s^3\psi_e c\phi_e - s^2\psi_e s\phi_e s\theta_e c\psi_e - s\psi_e c^2\psi_e c\phi_e + c^3\psi_e s\theta_e s\phi_e)s^2\beta_{1c}c\gamma_{1c}c\beta_{1c} + (c\theta_e)(s^2\psi_e c\psi_e c\phi_e + s^3\psi_e s\theta_e s\phi_e - c^3\psi_e c\phi_e - s\psi_e s\theta_e s\phi_e c^2\psi_e)c\gamma_{1c}c^2\beta_{1c}s\beta_{1c}$$

$$- (c^2\theta_e s\phi_e)(s^2\psi_e - c^2\psi_e)s\gamma_{1c}s\beta_{1c}c^2\beta_{1c} + (s^2\psi_e c\psi_e c\theta_e c\phi_e - s\psi_e s\theta_e s\phi_e c^2\psi_e c\theta_e)s\beta_{1c}c^2\gamma_{1c}c^2\beta_{1c} - (s\psi_e c^2\psi_e c\theta_e c\phi_e + s^2\psi_e s\theta_e s\phi_e c\psi_e c\theta_e)c^3\gamma_{1c}c^3\beta_{1c}$$

$$+ (s\psi_e s\phi_e c\psi_e c^2\theta_e)s\gamma_{1c}c^3\beta_{1c}c^2\gamma_{1c} - (s^2\psi_e s\theta_e c\phi_e - s\psi_e s^2\theta_e s\phi_e c\psi_e)s^2\beta_{1c}s\gamma_{1c}c\beta_{1c} + (s\psi_e s\theta_e c\psi_e c\phi_e + s^2\psi_e s^2\theta_e s\phi_e)c\gamma_{1c}c^2\beta_{1c}s\beta_{1c}s\gamma_{1c}$$

$$- (s\psi_e s\theta_e s\phi_e c\theta_e)s^2\gamma_{1c}s\beta_{1c}c^2\beta_{1c} + (s\psi_e s\theta_e c\psi_e c\phi_e - c^2\psi_e c\theta_e c\phi_e)s\beta_{1c}s\gamma_{1c}c\gamma_{1c}c^2\beta_{1c} + (c^2\psi_e c\phi_e s\theta_e + s\psi_e s^2\theta_e s\phi_e c\psi_e)c^2\gamma_{1c}c^3\beta_{1c}s\gamma_{1c}$$

$$+ (c\psi_e c\theta_e s\theta_e s\phi_e)s^2\gamma_{1c}c^3\beta_{1c}c\gamma_{1c}$$

$$\delta_{18} = \delta_{10}$$

$$\nu_{19} = -(s\phi_e s\theta_e)s\beta_{1c}c\gamma_{2c}c\beta_{2c} - (s\phi_e s\theta_e)s\beta_{2c}c\gamma_{1c}c\beta_{1c} + (c\theta_e s\psi_e s\phi_e)s\beta_{1c}s\gamma_{2c}c\beta_{2c} + (c\theta_e s\psi_e s\phi_e)s\beta_{2c}s\gamma_{1c}c\beta_{1c}$$

$$- (c\theta_e c\psi_e s\phi_e)c\gamma_{1c}c\beta_{1c}s\gamma_{2c}c\beta_{2c} + (c\theta_e c\psi_e s\phi_e)c\gamma_{2c}c\beta_{2c}s\gamma_{1c}c\beta_{1c}$$

$$\delta_{19} = (\delta_{10})^{1/3}(\delta_9)^{1/3}$$

$$A_{84} = \frac{K_m K_c}{\tau_m}(-d_{12})(1 + \frac{\tau_1}{\tau_2})(\Delta\beta_1^e \frac{\nu_{17}}{\delta_{17}} + \Delta\beta_2^e \frac{\nu_{18}}{\delta_{18}} + \frac{\nu_{19}}{\delta_{19}})$$

$$\nu_{20} = -(s\psi_e s\theta_e c\theta_e - c\psi_e s^2\theta_e s\theta_e)s^3\beta_{2c} - (c\psi_e c\theta_e s\theta_e + s\psi_e s^2\theta_e s\theta_e)s^2\beta_{2c}c\gamma_{2c}c\beta_{2c} + (s\theta_e s\theta_e c\theta_e)s^2\beta_{2c}s\gamma_{2c}c\beta_{2c}$$

$$- (s\psi_e s\theta_e c\theta_e - c\psi_e s\theta_e s\theta_e)s\beta_{2c}c^2\gamma_{2c}c^2\beta_{2c} - (c\psi_e c\theta_e c\psi_e + s\psi_e s^2\theta_e s\theta_e)c^3\gamma_{2c}c^3\beta_{2c} + (s\theta_e s\theta_e c\theta_e)s\gamma_{2c}c^2\gamma_{2c}c^3\beta_{2c}$$

$$- (s\psi_e c\psi_e c\theta_e c\theta_e - c^2\psi_e c\theta_e s\theta_e s\theta_e)s^2\beta_{2c}s\gamma_{2c}c\beta_{2c} - (c^2\psi_e c\theta_e c\theta_e + s\psi_e s\theta_e s\theta_e c\psi_e c\theta_e)s\gamma_{2c}s\beta_{2c}c^2\beta_{2c}c\gamma_{2c}$$

$$+ (c\psi_e c^2\theta_e s\psi_e)s^2\gamma_{2c}s\beta_{2c}c^2\beta_{2c} + (s^2\psi_e c\theta_e c\theta_e - s\psi_e s\theta_e s\theta_e c\psi_e c\theta_e)s\gamma_{2c}s\beta_{2c}c^2\beta_{2c}c\gamma_{2c}$$

$$+ (s\psi_e c\psi_e c\theta_e c\theta_e + s^2\psi_e s\theta_e s\theta_e c\theta_e)c^2\gamma_{2c}c^3\beta_{2c}s\gamma_{2c} - (s\psi_e s\theta_e c^2\theta_e)s^2\gamma_{2c}c^3\beta_{2c}c\gamma_{2c}$$

$$\delta_{20} = \delta_9$$

Fig. A-7 (Cont) The Elements of Matrix A

$$\nu_{21} = (s\psi_e\, s\theta_e\, c\phi_e - c\psi_e\, s^2\theta_e\, s\phi_e)\, s^3\beta_{1c} - (c\psi_e\, c\phi_e\, s\theta_e + s\psi_e\, s^2\theta_e\, s\phi_e)\, s^2\beta_{1c}\, c\gamma_{1c}\, c\beta_{1c} + (s\theta_e\, s\phi_e\, c\theta_e)\, s^2\beta_{1c}\, s\gamma_{1c}\, c\beta_{1c}$$

$$+ (s\psi_e\, s\theta_e\, c\phi_e - c\psi_e\, s\theta_e\, s\phi_e)\, s\beta_{1c}\, c^2\gamma_{1c}\, c^2\beta_{1c} - (c\psi_e\, c\theta_e\, c\psi_e + s\psi_e\, s^2\theta_e\, s\phi_e)\, c^3\gamma_{1c}\, c^3\beta_{1c} + (s\theta_e\, s\phi_e\, c\theta_e)\, s\gamma_{1c}\, c^2\gamma_{1c}\, c^3\beta_{1c}$$

$$- (s\psi_c\, c\psi_e\, c\theta_e\, c\phi_e - c^2\psi_e\, c\theta_e\, s\theta_e\, s\phi_e)\, s^2\beta_{1c}\, s\gamma_{1c}\, c\beta_{1c} + (c^2\psi_e\, c\theta_e\, c\phi_e + s\psi_e\, s\theta_e\, s\phi_e\, c\psi_e\, c\theta_e)\, s\gamma_{1c}\, s\beta_{1c}\, c^2\beta_{1c}\, c\gamma_{1c}$$

$$- (c\psi_e\, c^2\theta_e\, s\psi_e)\, s^2\gamma_{1c}\, s\beta_{1c}\, c^2\beta_{1c} - (s^2\psi_e\, c\theta_e\, c\phi_e - s\psi_e\, s\theta_e\, s\phi_e\, c\psi_c\, c\theta_e)\, s\gamma_{1c}\, s\beta_{1c}\, c^2\beta_{1c}\, c\gamma_{1c}$$

$$+ (s\psi_e\, c\psi_e\, c\theta_e\, c\phi_e + s^2\psi_e\, s\theta_e\, s\phi_e\, c\theta_e)\, c\gamma_{1c}\, c^3\beta_{1c}\, s\gamma_{1c} - (s\psi_e\, s\phi_e\, c^2\theta_e)\, s^2\gamma_{1c}\, c^3\beta_{1c}\, c\gamma_{1c}$$

$$\iota_{21} = \iota_{10}$$

$$\nu_{22} = (s\theta_e\, c\phi_e\, c\theta_e)\, s\beta_{1c}\, c\gamma_{2c}\, c\beta_{2c} + (s\theta_e\, c\phi_e\, c\theta_e)\, s\beta_{2c}\, c\gamma_{1c}\, c\beta_{1c} - (c^2\theta_e\, c\phi_e\, c\psi_e)\, s\gamma_{1c}\, c\beta_{1c}\, c\gamma_{2c}\, c\beta_{1c}$$

$$+ (c^2\theta_e\, c\phi_e\, c\psi_e)\, s\gamma_{2c}\, c\beta_{2c}\, c\gamma_{1c}\, c\beta_{1c} - (c\psi_e\, c\phi_e\, c^2\theta_e)\, s\gamma_{1c}\, c\beta_{1c}\, s\beta_{2c} - (s\psi_e\, c\phi_e\, c^2\theta_e)\, s\gamma_{2c}\, c\beta_{2c}\, s\beta_{1c}$$

$$\iota_{22} = (\iota_{10})^{1/3}\,(\iota_9)^{1/3}$$

$$A_{87} = \frac{-K_m K_c}{\tau_m}\left(1 + \frac{\tau_1}{\tau_2}\right)(-d_{12})\left(\Delta\beta_1^e \frac{\nu_{20}}{\iota_{20}} + \Delta\beta_2^e \frac{\nu_{21}}{\iota_{21}} + \frac{\nu_{22}}{\iota_{22}}\right)$$

$$A_{91} = \frac{-K_c \tau_1}{\tau_2^2}(-d_{12})\left(\Delta\beta_1^e \frac{\nu_{15}}{\iota_{15}} + \Delta\beta_2^e \frac{\nu_{16}}{\iota_{16}}\right)$$

$$A_{94} = \frac{-k_c \tau_1}{\tau_2^2}(-d_{12})\left(\Delta\beta_1 \frac{\nu_{17}}{\iota_{17}} + \Delta\beta_2 \frac{\nu_{18}}{\iota_{18}} + \frac{\nu_{19}}{\iota_{19}}\right)$$

$$A_{97} = \frac{-k_c \tau_1}{\tau_2^2}(-d_{12})\left(\Delta\beta_1 \frac{\nu_{20}}{\iota_{20}} + \Delta\beta_2 \frac{\nu_{21}}{\iota_{21}} + \frac{\nu_{22}}{\iota_{22}}\right)$$

Fig. A-7 (Cont) The Elements of Matrix A

A_{54}: $s_9 = (N_2^2 + D_2^2)^{3/2}$

$s_{10} = (N_1^2 + D_1^2)^{3/2}$

$s_{11} = (N_1^2 + D_1^2)^{1/2} (N_2^2 + D_2^2)^{1/2} = (s_{10})^{1/3} (s_9)^{1/3}$

A_{81}: $s_{15} = (N_2^2 + D_2^2)^{1/2} = (s_9)^{1/3}$

$s_{16} = (N_1^2 + D_1^2)^{1/2} = (s_{10})^{1/3}$

A_{84}: $s_{17} = (N_2^2 + D_2^2)^{3/2} = s_9$

$s_{18} = (N_1^2 + D_1^2)^{3/2} = s_{10}$

$s_{19} = (N_1^2 + D_1^2)^{1/2} (N_2^2 + D_2^2)^{1/2} = (s_{10})^{1/3} (s_9)^{1/3}$

A_{87}: $s_{20} = (N_2^2 + D_2^2)^{3/2} = s_9$

$s_{21} = (N_1^2 + D_1^2)^{3/2} = s_{10}$

$s_{22} = (N_1^2 + D_1^2)^{1/2} (N_2^2 + D_2^2)^{1/2} = (s_{10})^{1/3} (s_9)^{1/3}$

Fig. A-7 (Cont) The Elements of Matrix A

$$\tilde{A}_{ij}(\theta_e,\phi_e,\psi_e) = A_{ij}(\underline{0}) + \frac{\partial A_{ij}}{\partial \theta_e}\Big|_{\underline{0}} \cdot \theta_e + \frac{\partial A_{ij}}{\partial \phi_e}\Big|_{\underline{0}} \cdot \phi_e + \frac{\partial A_{ij}}{\partial \psi_e}\Big|_{\underline{0}} \cdot \psi_e \qquad \text{(A-47)}$$

where $\underline{0} = (0,0,0)$. The results of this approximation are given in Fig. A-6b.

If the effects of nonzero equilibrium are assumed to be negligibly small (something that remains to be proved), then the equations of Fig. A-6a become those of Fig. A-8 where the variables v', ω' have been redefined to be dimensionless, i.e.,

$$v''_\theta = \frac{1}{K_m K_c} v'_\theta \ , \quad \text{etc.}$$

$$\omega''_\theta = \frac{\tau_2}{K_c \tau_1} \omega'_\theta \ , \quad \text{etc.} \qquad \text{(A-48)}$$

Note that it is in this simplified form that the decoupling feature of the "paired tracker" design is evident.

The state equations of the model based on the tracker rate model are

$$\dot{v}'_\phi = -\frac{1}{\tau_m} v'_\phi + \frac{K_m}{\tau_m} f\left(K_c\left(1+\frac{\tau_1}{\tau_2}\right)\epsilon'_\phi + \omega_\phi + \frac{I}{K_m} h^o_\phi\right) - \frac{I}{\tau_m} h^o_\phi$$

$$\dot{\omega}'_\phi = -\frac{1}{\tau_2}\omega'_\phi - \frac{K_c \tau_1}{\tau_2^2}\epsilon'_\phi$$

$$\dot{v}'_\theta = -\frac{1}{\tau_m} v'_\theta + \frac{K_m}{\tau_m} f\left(K_c\left(1+\frac{\tau_1}{\tau_2}\right)\epsilon'_\theta + \omega'_\theta + \frac{I}{K_m} h^o_\theta\right) - \frac{I}{\tau_m} h^o_\theta \qquad \text{(A-49)}$$

$$\dot{\omega}'_\theta = -\frac{1}{\tau_2}\omega'_\theta - \frac{K_c \tau_1}{\tau_2^2}\epsilon'_\theta$$

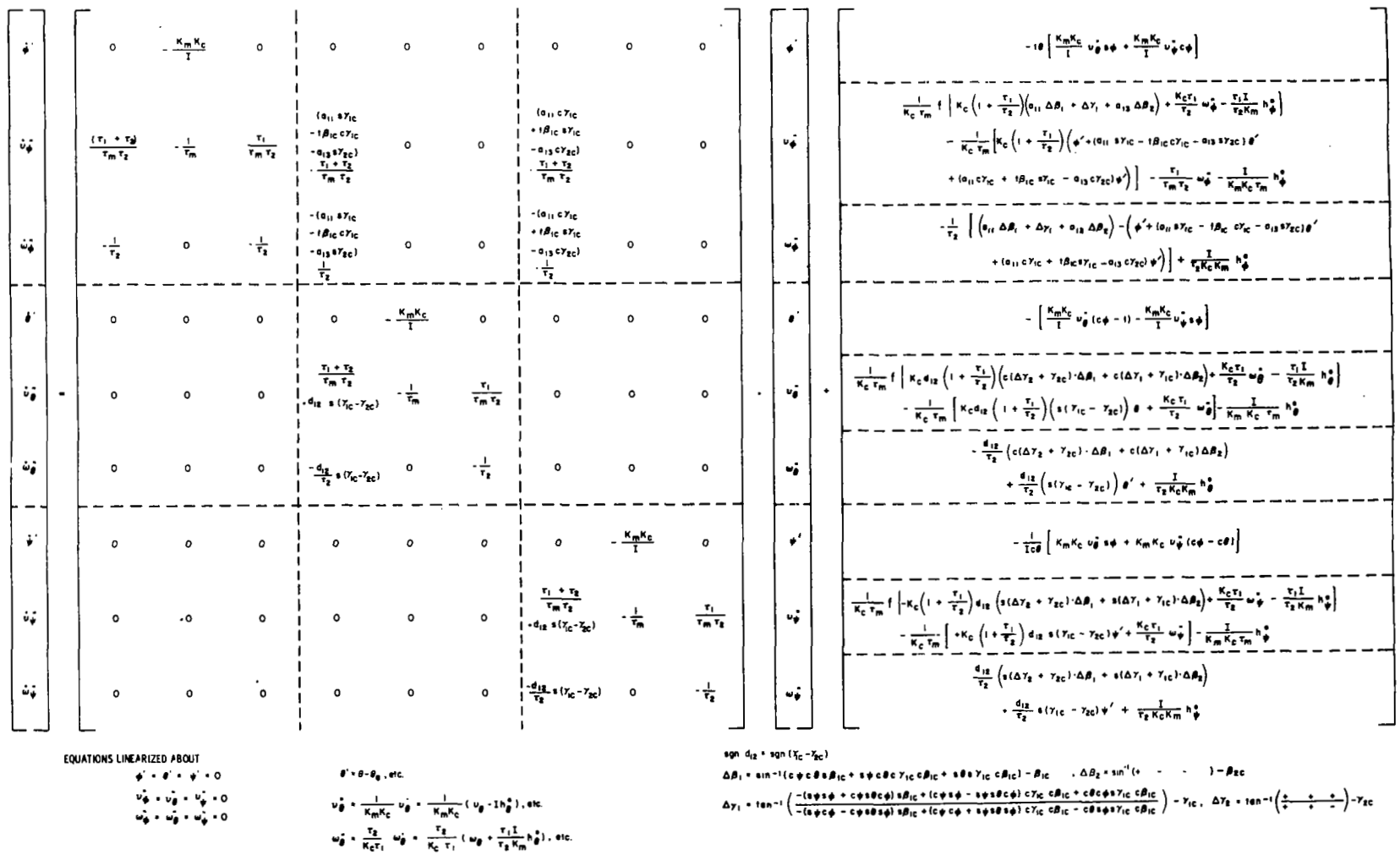


Fig. A-8 Nondimensional State Equations Based on Tracker Angle Model - Offset Neglected

$$\dot{v}'_\psi = -\frac{1}{\tau_m} v'_\psi + \frac{K_m}{\tau_m} f\left(K_c\left(1 + \frac{\tau_1}{\tau_2}\right)\epsilon'_\psi + \omega'_\psi + \frac{I}{K_m} h^o_\psi\right) - \frac{I}{\tau_m} h^o_\psi$$

$$\dot{\omega}'_\psi = -\frac{1}{\tau_2} \omega'_\psi - \frac{K_c \tau_1}{\tau_2^2} \epsilon'_\psi$$

$$\epsilon'_\phi = a_{11}\Delta\beta_1 + \Delta\gamma_1 + a_{13}\Delta\beta_2 - \frac{I}{K_c K_m} h^o_\phi$$

$$\epsilon'_\theta = d_{12}c(\gamma_{2c} + \Delta\gamma_2) \cdot \Delta\beta_1 + d_{12}c(\gamma_{1c} + \Delta\gamma_1) \cdot \Delta\beta_2 - \frac{I}{K_c K_m} h^o_\theta \qquad \text{(A-49) (Cont.)}$$

$$\epsilon'_\psi = -d_{12}s(\gamma_{2c} + \Delta\gamma_2) \cdot \Delta\beta_1 - d_{12}s(\gamma_{1c} + \Delta\gamma_1) \cdot \Delta\beta_2 - \frac{I}{K_c K_m} h^o_\psi$$

$$\dot{\Delta\beta}_1 = -s(\Delta\gamma_1 + \gamma_{1c}) \frac{v'_\theta}{I} - c(\Delta\gamma_1 + \gamma_{1c}) \frac{v'_\psi}{I}$$

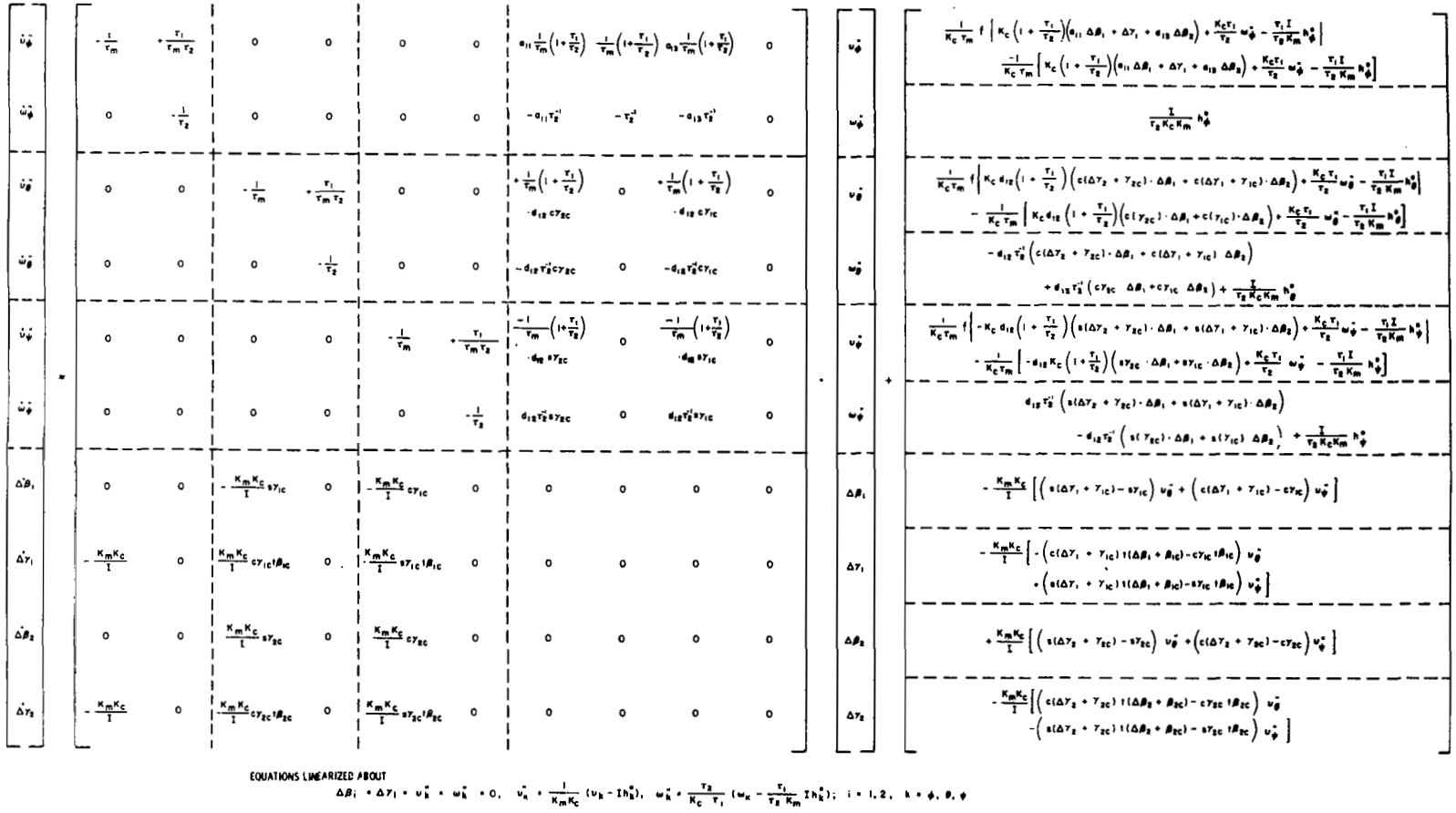
$$\dot{\Delta\gamma}_1 = -\frac{v'_\phi}{I} + t(\Delta\beta_1 + \beta_{1c})c(\Delta\gamma_1 + \gamma_{1c}) \frac{v'_\theta}{I} + t(\Delta\beta_1 + \beta_{1c})s(\Delta\gamma_1 + \gamma_{1c}) \frac{v'_\psi}{I}$$

$$\dot{\Delta\beta}_2 = s(\Delta\gamma_2 + \gamma_{2c}) \frac{v'_\theta}{I} + c(\Delta\gamma_2 + \gamma_{2c}) \frac{v'_\psi}{I}$$

$$\dot{\Delta\gamma}_2 = -\frac{v'_\phi}{I} - t(\Delta\beta_2 + \beta_{2c})c(\Delta\gamma_2 + \gamma_{2c}) \frac{v'_\theta}{I} - t(\Delta\beta_2 + \beta_{2c})s(\Delta\gamma_2 + \gamma_{2c}) \frac{v'_\psi}{I} .$$

These equations are linearized exactly the way Eqs. (A-40) were and the result, assuming nonzero equilibrium effects to be negligible, is given in Fig. A-9. Note that there are ten state variables as opposed to nine for the tracker angle model, the decoupling is not evident, and the matrix A has one zero eigenvalue (due to the column of zeros). Since the stability analysis requires that the linear part of the system be asymptotically stable, this model will not be considered further.

The model based on the tracker angle model in the simplified form (assuming zero offset effects) will be the only one whose stability is studied herein. Its relationship to the model analyzed and simulated by Doolin and Showman [1] and [2] can be seen by



EQUATIONS LINEARIZED ABOUT

$$\Delta\beta_i = \Delta\gamma_i = v_h^* = \omega_h^* = 0, \quad v_h^* = \frac{1}{K_m K_c}(v_h - I h_h^*), \quad \omega_h^* = \frac{\tau_2}{K_c \tau_1}\left(\omega_h - \frac{\tau_1}{\tau_2 K_m} I h_h^*\right); \quad i = 1,2, \quad h = \phi, \theta, \psi$$

Fig. A-9 Nondimensional State Equations Based on Tracker Rate Model

recognizing that they study only the linear model. Their model is obtained by neglecting the $\Delta\gamma_1$, $\Delta\gamma_2$ variations in the processor, linearizing the function $f(\cdot)$, and linearizing the tracker rate model. The latter step is carried out by turning the tracker differential equations into incremental equations by multiplying through by an infinitesimal time Δt, using the approximations

$$\begin{aligned} p\Delta t &\approx \Delta\phi \\ q\Delta t &\approx \Delta\theta \\ r\Delta t &\approx \Delta\psi \end{aligned} \tag{A-50}$$

and ignoring the variations $\Delta\beta$, $\Delta\gamma$ in the right hand side. One then obtains the approximate gimbal angle errors $\Delta\tilde{\beta}$, $\Delta\tilde{\gamma}$

$$\begin{aligned} \Delta\tilde{\beta}_1 &= (s\gamma_{1c})\Delta\theta + (c\gamma_{1c})\Delta\psi \\ \Delta\tilde{\gamma}_1 &= \Delta\phi - (t\beta_{1c}c\gamma_{1c})\Delta\theta + (t\beta_{1c}s\gamma_{1c})\Delta\psi \\ \Delta\tilde{\beta}_2 &= -(s\gamma_{2c})\Delta\theta - (c\gamma_{2c})\Delta\psi \\ \Delta\tilde{\gamma}_2 &= \Delta\phi + (t\beta_{2c}c\gamma_{2c})\Delta\theta - (t\beta_{2c}s\gamma_{2c})\Delta\psi \ . \end{aligned} \tag{A-51}$$

It turns out that (A-51) represents the first order terms of a power series expansion of the tracker angle model equations about $\phi = \theta = \psi = 0$.

The Popov Approximation

An analysis of the simplified system where the only nonlinearities considered are those of motor saturation has been carried out literally because the numerical computation of the coefficient matrices was too sensitive. The simplified system is represented as

$$\dot{x} = Ax + Gf^a(u) \ , \tag{A-52}$$

where G is a 9 x 3 matrix, $f^a(u)$ is a three vector of saturation functions obtained from g(x) by deleting all nonlinear terms except saturation and linearizing the arguments of the saturations, and u is a three dimensional vector. These terms are:

$$A = \begin{bmatrix}
0 & -\frac{K_m K_c}{I} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -\frac{1}{\tau_m} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-\frac{1}{\tau_2} & 0 & -\frac{1}{\tau_2} & 2t\beta_{1c}c\gamma_{1c} & 0 & 0 & -2t\beta_{1c}s\gamma_{1c} & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{K_m K_c}{I} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{1}{\tau_m} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{1}{\tau_2} & 0 & -\frac{1}{\tau_2} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{K_m K_c}{I} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau_m} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau_2} & 0 & -\frac{1}{\tau_2}
\end{bmatrix}$$

$$G = \begin{bmatrix}
0 & 0 & 0 \\
\frac{1}{K_c \tau_m} & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & \frac{1}{K_c \tau_m} & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & \frac{1}{K_c \tau_m}
\end{bmatrix} \qquad \text{A-53)}$$

$$x^T = \left[\phi', v''_\phi, \omega'''_\phi, \theta', v''_\theta, \omega'''_\theta, \psi', v''_\psi, \omega'''_\psi\right]$$

$$\omega'''_\theta = \frac{\omega''_\theta}{d_{12}s(\gamma_{1c} - \gamma_{2c})} \quad ; \quad \omega'''_\psi = \frac{\omega''_\psi}{d_{12}s(\gamma_{1c} - \gamma_{2c})}$$

$$f^a(u) = \begin{bmatrix} f\left\{K_c\left(1+\frac{\tau_1}{\tau_2}\right)(\phi' - t\beta_{1c}c\gamma_{1c}\theta' + t\beta_{1c}s\gamma_{1c}\psi') + \frac{K_c\tau_1}{\tau_2} d_{12}s(\gamma_{1c}-\gamma_{2c})\omega_\phi'' + \frac{I}{K_m} h_\phi^o\right\} - \frac{I}{K_m} h_\phi^o \\ f\left\{K_c\left(1+\frac{\tau_1}{\tau_2}\right)d_{12}\left[s(\gamma_{1c}-\gamma_{2c})\cdot\theta'\right] + \frac{K_c\tau_1}{\tau_2} d_{12}s(\gamma_{1c}-\gamma_{2c})\omega_\theta'' + \frac{I}{K_m} h_\theta^o\right\} - \frac{I}{K_m} h_\theta^o \\ f\left\{K_c\left(1+\frac{\tau_1}{\tau_2}\right)d_{12}\left[s(\gamma_{1c}-\gamma_{2c})\cdot\psi'\right] + \frac{K_c\tau_1}{\tau_2} d_{12}s(\gamma_{1c}-\gamma_{2c})\omega_\psi'' + \frac{I}{K_m} h_\psi^o\right\} - \frac{I}{K_m} h_\psi^o \end{bmatrix}$$

or

$$f^a(u) = \begin{bmatrix} f\left\{u_\phi + \frac{I}{K_m} h_\phi^o\right\} - \frac{I}{K_m} h_\phi^o \\ f\left\{u_\theta + \frac{I}{K_m} h_\theta^o\right\} - \frac{I}{K_m} h_\theta^o \\ f\left\{u_\psi + \frac{I}{K_m} h_\psi^o\right\} - \frac{I}{K_m} h_\psi^o \end{bmatrix}$$

If we define T to be a matrix whose columns are the eigenvectors of A, and relate y to x by

$$x = Ty \tag{A-54}$$

we obtain

$$\dot{y} = T^{-1}ATy + T^{-1}Gf^a(u) \ . \tag{A-55}$$

It can be shown that

$$T^{-1}AT = \text{diag}\left[0,\ 0,\ 0,\ -\frac{1}{\tau_m},\ -\frac{1}{\tau_m},\ -\frac{1}{\tau_m},\ -\frac{1}{\tau_2},\ -\frac{1}{\tau_2},\ -\frac{1}{\tau_2}\right] , \tag{A-56}$$

where T can be written as

$$T = \begin{vmatrix} 1 & 0 & (2\tau_2 t\beta_{1c} c\gamma_{1c}) & \frac{\tau_2}{\tau_m} - 1 & 0 & -2\tau_2 t\beta_{1c} s\gamma_{1c} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{I(\tau_2 - \tau_m)}{K_m K_c \tau_m^2} & 0 & \frac{-2I\tau_2 t\beta_{1c} s\gamma_{1c}}{\tau_m K_m K_c} & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & \frac{-2\tau_2 \tau_m t\beta_{1c} c\gamma_{1c}}{\tau_2 - \tau_m} & 0 & 1 & 0 & 0 \\ 0 & \frac{-1}{c\gamma_{1c}} & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{I}{\tau_m K_m K_c} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{c\gamma_{1c}} & -1 & 0 & \frac{\tau_m}{\tau_2 - \tau_m} & 0 & 0 & 1 & 0 \\ 0 & \frac{-1}{s\gamma_{1c}} & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{I}{\tau_m K_m K_c} & 0 & 0 & 0 \\ 0 & \frac{1}{s\gamma_{1c}} & 0 & 0 & 0 & -\frac{\tau_m}{\tau_m - \tau_2} & 0 & 0 & 1 \end{vmatrix} \quad \text{(A-57)}$$

with

$$T^{-1} = \begin{vmatrix} 1 & -\frac{K_m K_c \tau_m}{I} & 0 & -2\tau_2 t\beta_{1c} c\gamma_{1c} & \frac{2\tau_2 \tau_m K_m K_c t\beta_{1c} c\gamma_{1c}}{I} & 0 & 2\tau_2 t\beta_{1c} s\gamma_{1c} & -\frac{2\tau_2 \tau_m K_m K_c t\beta_{1c} s\gamma_{1c}}{I} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -s\gamma_{1c} & \frac{K_m K_c \tau_m s\gamma_{1c}}{I} & 0 \\ 0 & 0 & 0 & 1 & -\frac{K_m K_c \tau_m}{I} & 0 & -t\gamma_{1c} & \frac{K_m K_c \tau_m s\gamma_{1c}}{Ic\gamma_{1c}} & 0 \\ 0 & \frac{K_m K_c \tau_m^2}{I(\tau_2 - \tau_m)} & 0 & 0 & 0 & 0 & 0 & \frac{2\tau_2 \tau_m^2 K_m K_c t\beta_{1c} s\gamma_{1c}}{I(\tau_2 - \tau_m)} & 0 \\ 0 & 0 & 0 & 0 & \frac{K_m K_c \tau_m}{I} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{K_m K_c \tau_m}{I} & 0 \\ 1 & \frac{\tau_m \tau_2 K_m K_c}{(\tau_m - \tau_2)I} & 1 & -2\tau_2 t\beta_{1c} c\gamma_{1c} & \frac{2K_m K_c \tau_2^2 t\beta_{1c} c\gamma_{1c} \tau_m}{I(\tau_2 - \tau_m)} & 0 & 2\tau_2 t\beta_{1c} s\gamma_{1c} & -\frac{2\tau_2^2 \tau_m K_m K_c t\beta_{1c} s\gamma_{1c}}{I(\tau_2 - \tau_m)} & 0 \\ 0 & 0 & 0 & 1 & \frac{\tau_m K_m K_c \tau_2}{I(\tau_m - \tau_2)} & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -\frac{\tau_2 \tau_m K_m K_c}{I(\tau_2 - \tau_m)} & 1 \end{vmatrix} \quad \text{(A-58)}$$

If we now proceed to "tear" the system by defining

$$z^T = [y_4,\ y_5,\ y_6,\ y_7,\ y_8,\ y_9] \tag{A-59}$$

where superscript T denotes transpose, it can be shown that the system equations will reduce to the form

$$\begin{aligned} \dot{z} &= A^* z + B^* f^a(u) \\ \dot{u} &= H^* z + J^* f^a(u)\ , \end{aligned} \tag{A-60}$$

where

$$J^* \equiv 0$$

$$A^* = \text{diag}\left[-\frac{1}{\tau_m},\ -\frac{1}{\tau_m},\ -\frac{1}{\tau_m},\ -\frac{1}{\tau_2},\ -\frac{1}{\tau_2},\ -\frac{1}{\tau_2} \right]$$

$$B^* = \begin{bmatrix} \frac{K_m \tau_m}{I(\tau_2 - \tau_m)} & 0 & \frac{2\tau_2 \tau_m K_m t\beta_{1c} s\gamma_{1c}}{I(\tau_2 - \tau_m)} \\ 0 & \frac{K_m}{I} & 0 \\ 0 & 0 & \frac{K_m}{I} \\ \frac{K_m \tau_2}{I(\tau_m - \tau_2)} & \frac{2K_m \tau_2^2 t\beta_{1c} c\gamma_{1c}}{I(\tau_2 - \tau_m)} & \frac{-2\tau_2^2 K_m t\beta_{1c} s\gamma_{1c}}{I(\tau_2 - \tau_m)} \\ 0 & \frac{K_m \tau_2}{I(\tau_m - \tau_2)} & 0 \\ 0 & 0 & \frac{K_m \tau_2}{I(\tau_m - \tau_2)} \end{bmatrix} \tag{A-61}$$

$$H^* = \begin{bmatrix} \frac{K_c}{\tau_m}\left[1 - \frac{\tau_2 + \tau_1}{\tau_m}\right] & \frac{K_c t\beta_{1c} c\gamma_{1c}}{\tau_m}\left[\frac{2\tau_1\tau_m}{\tau_2 - \tau_m} + \frac{\tau_2 + \tau_1}{\tau_2}\right] & \frac{K_c}{\tau_m}\left(1 + \frac{\tau_1}{\tau_2}\right)(2\tau_2 - 1)t\beta_{1c} s\gamma_{1c} & -\frac{K_c\tau_1}{\tau_2^2} & 0 & 0 \\ 0 & -\frac{K_c}{\tau_m} d_{12} s(\gamma_{1c} - \gamma_{2c})\left[\frac{\tau_1 + \tau_2 - \tau_m}{\tau_2 - \tau_m}\right] & 0 & 0 & -\frac{K_c\tau_1}{\tau_2^2} d_{12} s(\gamma_{1c} - \gamma_{2c}) & 0 \\ 0 & 0 & -\frac{K_c}{\tau_m} d_{12} s(\gamma_{1c} - \gamma_{2c})\left[\frac{\tau_1 + \tau_2 - \tau_m}{\tau_2 - \tau_m}\right] & 0 & 0 & -\frac{K_c\tau_1}{\tau_2^2} d_{12} s(\gamma_{1c} - \gamma_{2c}) \end{bmatrix}$$

Taking the Laplace transform of (A-60) results in the system

$$u(s) = \frac{1}{s} H^*(sI - A^*)B^* f^a(u) = - W(s) f^a(u) \qquad \text{(A-62)}$$

where

$$W(s) = Ww(s)$$

$$w(s) = \frac{K_m K_c}{I} \frac{(\tau_1 + \tau_2)s + 1}{s(\tau_m s + 1)(\tau_2 s + 1)} \qquad \text{(A-63)}$$

$$W = \begin{bmatrix} 1 & -t\beta_{1c} c\gamma_{1c} & t\beta_{1c} s\gamma_{1c} \\ 0 & d_{12} s(\gamma_{1c} - \gamma_{2c}) & 0 \\ 0 & 0 & d_{12} s(\gamma_{1c} - \gamma_{2c}) \end{bmatrix}$$

This is a Popov type model with the multiple nonlinearities expressed by the vector $f^a(u)$ where the elements are given by

$$f_i^a(u) = 26\ \text{sat}\left[\frac{1}{26}\left(u_i + \frac{Ih_i^o}{K_m}\right)\right] - \frac{Ih_i^o}{K_m}, \qquad i = \phi, \theta, \psi \qquad \text{(A-64)}$$

Elimination of Compensator Lag Dynamics

The transfer function of the lead-lag compensation network is described by

$$G_c(s) = K_c \frac{(\tau_1 + \tau_2)s + 1}{\tau_2 s + 1}, \qquad \text{(A-65)}$$

where

$$K_c = 2.685 \times 10^5 \text{ volt/rad}$$

$$\tau_1 = 4.5 \text{ sec} \quad , \quad \tau_2 = 0.5 \text{ sec} \; . \tag{A-66}$$

Therefore,

$$G_c(s) = K_c\left[\frac{9s}{s+2} + 1\right] \tag{A-67}$$

Considering the fact that the rotational rates of the vehicle are much slower than 2 rad/sec, let us ignore for this treatment the effect of lag dynamics in $G_c(s)$. With this simplifying assumption, the resulting equations become

$$\frac{V'(s)}{\epsilon(s)} = K_c(4.5\, s + 1) \tag{A-68}$$

with a corresponding differential equation for each channel,

$$V' = K_c(4.5\, \dot{\epsilon} + \epsilon) \; . \tag{A-69}$$

Proceeding to solve for ϵ, $\dot{\epsilon}$, we obtain:

$$\begin{aligned}
\epsilon_\phi &= \Delta\gamma_1 \\
\epsilon_\theta &= d_{12}\left[c(\Delta\gamma_2 + \gamma_{2c})\Delta\beta_1 + c(\Delta\gamma_1 + \gamma_{1c})\Delta\beta_2\right] \\
\epsilon_\psi &= -\, d_{12}\left[s(\Delta\gamma_2 + \gamma_{2c})\Delta\beta_1 + s(\Delta\gamma_1 + \gamma_{1c})\Delta\beta_2\right]
\end{aligned} \tag{A-70}$$

$$\begin{aligned}
\dot{\epsilon}_\phi = \dot{\Delta\gamma_1} = p &- t(\Delta\beta_1 + \beta_{1c}) \cdot c(\Delta\gamma_1 + \gamma_{1c}) \cdot q \\
&+ t(\Delta\beta_1 + \beta_{1c}) \cdot s(\gamma_1 + \gamma_{1c}) \cdot r
\end{aligned} \tag{A-71}$$

$$\dot{\epsilon}_\phi = \left(h_\phi^o - \frac{v_\phi}{I}\right) - t(\Delta\beta_1 + \beta_{1c}) \cdot c(\Delta\gamma_1 + \gamma_{1c}) \cdot \left(h_\theta^o - \frac{v_\theta}{I}\right)$$

$$+ t(\Delta\beta_1 + \beta_{1c}) \cdot s(\Delta\gamma_1 + \gamma_{1c}) \cdot \left(h_\psi^o - \frac{v_\psi}{I}\right) \qquad \text{(A-71)}$$

$$\dot{\epsilon}_\theta = d_{12}c(\Delta\gamma_2 + \gamma_{2c}) \cdot \Delta\dot{\beta}_1 - d_{12}s(\Delta\gamma_2 + \gamma_{2c}) \cdot \Delta\gamma_2\Delta\beta_1$$

$$+ d_{12}c(\Delta\gamma_1 + \gamma_{1c}) \cdot \Delta\dot{\beta}_2 - d_{12}s(\Delta\gamma_1 + \gamma_{1c}) \cdot \Delta\dot{\gamma}_1\Delta\beta_2$$

By defining the following variables as:

$$PJ_1 \equiv s(\Delta\gamma_1 + \gamma_{1c}) \cdot \left(h_\theta^o - \frac{v_\theta}{I}\right) + c(\Delta\gamma_1 + \gamma_{1c}) \cdot \left(h_\psi^o - \frac{v_\psi}{I}\right)$$

$$PJ_2 \equiv - s(\Delta\gamma_2 + \gamma_{2c}) \cdot \left(h_\theta^o - \frac{v_\theta}{I}\right) - c(\Delta\gamma_2 + \gamma_{2c}) \cdot \left(h_\psi^o - \frac{v_\psi}{I}\right)$$

$$PJ_3 \equiv \left(h_\phi^o - \frac{v_\phi}{I}\right) + t(\Delta\beta_2 + \beta_{2c}) \cdot c(\Delta\gamma_2 + \gamma_{2c}) \cdot \left(h_\theta - \frac{v_\theta}{I}\right)$$

$$- t(\Delta\beta_2 + \beta_{2c}) \cdot s(\Delta\gamma_2 + \gamma_{2c}) \cdot \left(h_\psi^o - \frac{v_\psi}{I}\right)$$

$$PJ_4 \equiv \left(h_\phi^o - \frac{v_\phi}{I}\right) - t(\Delta\beta_1 + \beta_{1c}) \cdot c(\Delta\gamma_1 + \gamma_{1c}) \cdot \left(h_\theta^o - \frac{v_\theta}{I}\right) \qquad \text{(A-72)}$$

$$+ t(\Delta\beta_1 + \beta_{1c}) \cdot s(\Delta\gamma_1 + \gamma_{1c}) \cdot \left(h_\psi^o - \frac{v_\psi}{I}\right)$$

$$\dot{\epsilon}_\phi = PJ_4$$

$$\dot{\epsilon}_\theta = d_{12}\Big[c(\Delta\gamma_2 + \gamma_{2c}) \cdot PJ_1 + c(\Delta\gamma_1 + \gamma_{1c}) \cdot PJ_2$$

$$- s(\Delta\gamma_2 + \gamma_{2c}) \cdot \Delta\beta_1 \cdot PJ_3 - s(\Delta\gamma_1 + \gamma_{1c}) \cdot \Delta\dot{\beta}_2 \cdot PJ_4\Big]$$

$$\dot{\epsilon}_\psi = -d_{12}\Big[s(\Delta\gamma_2 + \gamma_{2c}) \cdot PJ_1 + s(\Delta\gamma_1 + \gamma_{1c}) \cdot PJ_2 + c(\Delta\gamma_2 + \gamma_{2c}) \cdot \Delta\beta_1 \cdot PJ_3 + c(\Delta\gamma_1 + \gamma_{1c}) \cdot \Delta\beta_2 \cdot PJ_4\Big] \quad \text{(A-72) (Cont.)}$$

the resulting state equations become:

$$\begin{aligned}
\dot{\phi} &= h_\phi^o - \frac{1}{I} v_\phi + (t\theta s\phi) \cdot \left(h_\theta^o - \frac{1}{I} v_\theta\right) + (t\theta c\phi) \cdot \left(h_\psi^o - \frac{1}{I} v_\psi\right) \\
\dot{v}_\phi &= -\frac{1}{\tau_m} v_\phi + \frac{K_m}{\tau_m} f\left(K_c(\epsilon_\phi + 4.5\,\dot{\epsilon}_\phi)\right) \\
\dot{\theta} &= c\phi \cdot \left(h_\theta^o - \frac{1}{I} v_\theta\right) - s\phi \cdot \left(h_\psi^o - \frac{1}{I} v_\psi\right) \\
\dot{v}_\theta &= -\frac{1}{\tau_m} v_\theta + \frac{K_m}{\tau_m} f\left(K_c(\epsilon_\theta + 4.5\,\dot{\epsilon}_\theta)\right) \\
\dot{\psi} &= \left(\frac{s\phi}{c\theta}\right)\left(h_\theta^o - \frac{1}{I} v_\theta\right) + \left(\frac{c\phi}{c\theta}\right)\left(h_\psi^o - \frac{1}{I} v_\psi\right) \\
\dot{v}_\psi &= -\frac{1}{\tau_m} v_\psi + \frac{K_m}{\tau_m} f\left(K_c(\epsilon_\psi + 4.5\,\dot{\epsilon}_\psi)\right) .
\end{aligned} \quad \text{(A-73)}$$

Rescaling as before, we obtain

$$v'' = \frac{1}{K_m K_c} v' = \frac{1}{K_m K_c}(v - Ih^o)$$

$$\phi' = \phi - \phi_e \,, \quad \text{etc.},$$

with (A-73) becoming:

$$\begin{aligned}
\dot{\phi}' &= -\frac{K_m K_c}{I}\Big[v''_\phi + (t\theta s\phi)v''_\phi + (t\theta c\phi)v''_\psi\Big] \\
\dot{v}''_\phi &= -\frac{1}{\tau_m} v''_\phi + \frac{1}{K_c \tau_m} f\Big(K_c(\epsilon_\phi + 4.5\,\dot{\epsilon}_\phi)\Big) - \frac{I}{\tau_m K_m K_c} h_\phi^o
\end{aligned} \quad \text{(A-74)}$$

$$
\begin{aligned}
\dot{\theta}' &= -\frac{K_m K_c}{I}\left[c\phi \cdot v''_\theta - s\phi \cdot v''_\psi\right] \\
\dot{v}_\theta &= -\frac{1}{\tau_m} v''_\theta + \frac{1}{K_c \tau_m} f\left(K_c(\epsilon_\theta + 4.5\,\dot{\epsilon}_\theta)\right) - \frac{I}{\tau_m K_m K_c} h^o_\theta \\
\dot{\psi}' &= -\frac{K_m K_c}{I}\left(\frac{s\phi}{c\theta} \cdot v''_\theta + \frac{c\phi}{c\theta} \cdot v''_\psi\right) \\
\dot{v}''_\psi &= -\frac{1}{\tau_m} v''_\psi + \frac{1}{K_c \tau_m} f\left(K_c(\epsilon_\psi + 4.5\,\dot{\epsilon}_\psi)\right) - \frac{I}{\tau_m K_m K_c} h^o_\psi \,.
\end{aligned}
\qquad \text{(A-74) (Cont.)}
$$

By separating the linear and nonlinear terms in (A-74), putting them into the required form (A-41), and assuming that $\Delta\beta_1^e$ and $\Delta\beta_2^e$ are negligible, it can be shown that the six dimensional state equations take the form shown in Fig. A-10. If we wish to consider the six dimensional model, and include only the nonlinearity due to the motor saturation function, the form of Fig. A-10 reduces to that of Fig. A-11

$$
\Delta\beta_1^e = \left[\Delta\beta_1(\phi_e,\ \theta_e,\ \psi_e)\right], \quad \text{etc.}
$$

$$
\begin{bmatrix} \dot{\phi}' \\ \dot{v}'_{\phi} \\ \dot{\theta}' \\ \dot{v}'_{\theta} \\ \dot{\psi}' \\ \dot{v}'_{\psi} \end{bmatrix} = \begin{bmatrix} 0 & -\frac{K_m K_c}{I} & 0 & 0 & 0 & 0 \\ \frac{1}{\tau_m} & -\frac{1}{\tau_m}\left(1 + 4.5\frac{K_m K_c}{I}\right) & -\frac{t\beta_{1c} c\gamma_{1c}}{\tau_m} & 4.5\frac{K_m K_c}{\tau_m I}\cdot t\beta_{1c} c\gamma_{1c} & \frac{t\beta_{1c} s\gamma_{1c}}{\tau_m} & -\frac{4.5\,K_m K_c}{\tau_m I}\cdot t\beta_{1c} s\gamma_{1c} \\ 0 & 0 & 0 & -\frac{K_m K_c}{I} & 0 & 0 \\ 0 & 0 & \frac{d_{12}}{\tau_m}\cdot s(\gamma_{1c}-\gamma_{2c}) & \frac{1}{\tau_m}\left[1 + \frac{4.5\,K_m K_c d_{12}}{I}\cdot s(\gamma_{1c}-\gamma_{2c})\right] & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{K_m K_c}{I} \\ 0 & 0 & 0 & 0 & \frac{d_{12}}{\tau_m}\cdot s(\gamma_{1c}-\gamma_{2c}) & -\frac{1}{\tau_m}\left[1 + \frac{4.5\,K_m K_c d_{12}}{I}\cdot s(\gamma_{1c}-\gamma_{2c})\right] \end{bmatrix} \begin{bmatrix} \phi' \\ v'_{\phi} \\ \theta' \\ v'_{\theta} \\ \psi' \\ v'_{\psi} \end{bmatrix} + \begin{bmatrix} -\frac{K_m K_c}{I}(t\theta s\phi v'_{\theta} + t\theta c\phi v'_{\psi}) \\ \frac{1}{K_c\tau_m} f\left(K_c(\epsilon_{\phi} + 4.5\,\dot{\epsilon}_{\phi})\right) - \frac{Ih^o_{\phi}}{\tau_m K_m K_c} - \frac{1}{K_c\tau_m}\left\{\text{lin } f_2\right\} \\ -\frac{K_m K_c}{I}\left((c\phi - 1)v'_{\theta} - s\phi v'_{\psi}\right) \\ \frac{1}{K_c\tau_m} f\left(K_c(\epsilon_{\theta} + 4.5\,\dot{\epsilon}_{\theta})\right) - \frac{Ih^o_{\theta}}{\tau_m K_m K_c} - \frac{1}{K_c\tau_m}\left\{\text{lin } f_4\right\} \\ -\frac{K_m K_c}{I}\left(\frac{s\phi}{c\theta}v'_{\theta} + \left(\frac{c\phi}{c\theta} - 1\right)v'_{\psi}\right) \\ \frac{1}{K_c\tau_m} f\left(K_c(\epsilon_{\psi} + 4.5\,\dot{\epsilon}_{\psi})\right) - \frac{Ih^o_{\psi}}{\tau_m K_m K_c} - \frac{1}{K_c\tau_m}\left\{\text{lin } f_6\right\} \end{bmatrix}
$$

$$
\text{lin } f_2 = K_c\left[\left(\phi' - (t\beta_{1c} c\gamma_{1c})\theta' + (t\beta_{1c} s\gamma_{1c})\psi'\right) + \frac{I}{K_c K_m} h^o_{\phi} - \frac{4.5\,K_m K_c}{I}\left(v'_{\phi} - (t\beta_{1c} c\gamma_{1c})v'_{\theta} + t\beta_{1c} s\gamma_{1c})v'_{\psi}\right)\right]
$$

$$
\text{lin } f_4 = K_c d_{12}\left[s(\gamma_{1c} - \gamma_{2c})\cdot\theta + \frac{Ih^o_{\theta}}{K_c K_m} - \frac{4.5\,K_m K_c}{I}(c\gamma_{2c} s\gamma_{1c} - c\gamma_{1c} s\gamma_{2c})v'_{\theta}\right]
$$

$$
\text{lin } f_6 = -K_c d_{12}\left[-s(\gamma_{1c} - \gamma_{2c})\psi' + \frac{Ih^o_{\psi}}{K_c K_m} - \frac{4.5\,K_m K_c}{I}(s\gamma_{2c} c\gamma_{1c} - s\gamma_{1c} c\gamma_{2c})v'_{\psi}\right]
$$

$$
\epsilon_{\phi} = \Delta\gamma_1 \qquad \dot{\epsilon}_{\phi} = PJ_4
$$

$$
\epsilon_{\theta} = d_{12}\left[c(\Delta\gamma_c + \gamma_{2c})\cdot\Delta\beta_1 + c(\Delta\gamma_1 + \gamma_{1c})\cdot\Delta\beta_2\right] \qquad \dot{\epsilon}_{\theta} = d_{12}\left[c(\Delta\gamma_2 + \gamma_{2c})\cdot PJ_1 + c(\Delta\gamma_1 + \gamma_{1c})\cdot PJ_2 - s(\Delta\gamma_2 + \gamma_{2c})\cdot\Delta\beta_1\cdot PJ_3 - s(\Delta\gamma_1 + \gamma_{1c})\cdot\Delta\beta_2\cdot PJ_4\right]
$$

$$
\epsilon_{\psi} = -d_{12}\,s(\Delta\gamma_2 + \gamma_{2c})\cdot\Delta\beta_1 + s(\Delta\gamma_1 + \gamma_{1c})\cdot\Delta\beta_2\right] \qquad \dot{\epsilon}_{\psi} = -d_{12}\left[s(\Delta\gamma_2 + \gamma_{2c})\cdot PJ_1 + s(\Delta\gamma_1 + \gamma_{1c})PJ_2 + c(\Delta\gamma_2 + \gamma_{2c})\cdot\Delta\beta_1\cdot PJ_3 + c(\Delta\gamma_1 + \gamma_{1c})\cdot\Delta\beta_2\cdot PJ_4\right]
$$

Fig. A-10 Six Dimensional Approximation of OAO "Paired-Tracker" System Model

$$
\begin{bmatrix} \dot{\phi}' \\ \dot{v}''_{\phi} \\ \dot{\theta}' \\ \dot{v}''_{\theta} \\ \dot{\psi}' \\ \dot{v}''_{\psi} \end{bmatrix}
=
\begin{bmatrix}
0 & -\frac{K_m K_c}{I} & 0 & 0 & 0 & 0 \\
\frac{1}{\tau_m} & -\frac{1}{\tau_m}\left(1 + 4.5\frac{K_m K_c}{I}\right) & -\frac{t\beta_{1c}c\gamma_{1c}}{\tau_m} & 4.5\frac{K_m K_c}{\tau_m I}\cdot t\beta_{1c}c\gamma_{1c} & \frac{t\beta_{1c}s\gamma_{1c}}{\tau_m} & -\frac{4.5\,K_m K_c}{\tau_m I}\cdot t\beta_{1c}s\gamma_{1c} \\
0 & 0 & 0 & -\frac{K_m K_c}{I} & 0 & 0 \\
0 & 0 & \frac{d_{12}}{\tau_m}\cdot s(\gamma_{1c}-\gamma_{2c}) & \frac{1}{\tau_m}\left[1 + \frac{4.5\,K_m K_c d_{12}}{I}\cdot s(\gamma_{1c}-\gamma_{2c})\right] & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -\frac{K_m K_c}{I} \\
0 & 0 & 0 & 0 & \frac{d_{12}}{\tau_m}\cdot s(\gamma_{1c}-\gamma_{2c}) & -\frac{1}{\tau_m}\left[1 + \frac{4.5\,K_m K_c d_{12}}{I}\cdot s(\gamma_{1c}-\gamma_{2c})\right]
\end{bmatrix}
\begin{bmatrix} \phi' \\ v''_{\phi} \\ \theta' \\ v''_{\theta} \\ \psi' \\ v''_{\psi} \end{bmatrix}
+
\begin{bmatrix}
0 \\
\frac{1}{K_c\tau_m} f(\text{lin } f_2) - \frac{Ih^o_{\phi}}{\tau_m K_m K_c} - \frac{1}{K_c\tau_m}\left\{\text{lin } f_2\right\} \\
0 \\
\frac{1}{K_c\tau_m} f(\text{lin } f_4) - \frac{Ih^o_{\theta}}{\tau_m K_m K_c} - \frac{1}{K_c\tau_m}\left\{\text{lin } f_4\right\} \\
0 \\
\frac{1}{K_c\tau_m} f(\text{lin } f_6) - \frac{Ih^o_{\psi}}{\tau_m K_m K_c} - \frac{1}{K_c\tau_m}\left\{\text{lin } f_6\right\}
\end{bmatrix}
$$

$$
\text{lin } f_2 = K_c\left[\left(\phi' - (t\beta_{1c}c\gamma_{1c})\theta' + (t\beta_{1c}s\gamma_{1c})\psi'\right) + \frac{I}{K_c K_m}h^o_{\phi} - \frac{4.5\,K_m K_c}{I}\left(v''_{\phi} - (t\beta_{1c}c\gamma_{1c})v''_{\theta} + t\beta_{1c}s\gamma_{1c})v''_{\psi}\right)\right]
$$

$$
\text{lin } f_4 = K_c d_{12}\left[s(\gamma_{1c}-\gamma_{2c})\cdot\theta + \frac{Ih^o_{\theta}}{K_c K_m} - \frac{4.5\,K_m K_c}{I}(c\gamma_{2c}s\gamma_{1c} - c\gamma_{1c}s\gamma_{2c})v''_{\theta}\right]
$$

$$
\text{lin } f_6 = -K_c d_{12}\left[-s(\gamma_{1c}-\gamma_{2c})\psi' + \frac{Ih^o_{\psi}}{K_c K_m} - \frac{4.5\,K_m K_c}{I}(s\gamma_{2c}c\gamma_{1c} - s\gamma_{1c}c\gamma_{2c})v''_{\psi}\right]
$$

Fig. A-11 Six Dimensional Approximation of OAO "Paired Tracker" System Model with Motor Saturation Only

APPENDIX B

SIMULATION RESULTS

This appendix describes the simulation effort performed in support of the study. The type of simulation used and the results obtained are discussed.

The simulation uses a fourth order Runge-Kutta integration package with a 0.1 sec integration time step size. The simulation runs in 1/3 real time. The simulation required that the system be described in the basic state variable form

$$\dot{x}_g = f_g(x_i) \qquad i,j = 1, \ldots, 9 .$$

The program has a plotter option (CALCOMP), which was used to obtain the figures presented below. The flow chart is presented in Fig. B-1 and in Appendix G. This is the flow of the main routine only. The subroutine of chief interest is "AFX," which contains the state equations. It is basically AFX that was modified for the various models. Listings are available upon request from the Research Department, Grumman Aerospace Corporation, Bethpage, N.Y., 11714.

Figure B-2 outlines the initial run plan. Five sets of runs for various tracker cases with the same initial conditions on the state (except for changes in the total momentum Ih_1^o) are presented. The system parameters are also given. Figures B-3 and B-4 present the results of Runs No. 2 and No. 5.

For all runs of the initial set there was a basic similarity of performance. Some unexplained differences in the "6-D" model do exist. These differences are not consistent channel to channel or run to run but do indicate that the lag in some minor way affects performance.

The main concern as regards comparison, is the relation between "AN" (the exact model) and "MV," the model that is identical to the Doolin-Showman model in the feedback path. The cases considered initially, for which "AN" has good performance, show a nearly identical performance for "AN" and "MV." The only difference occurred in the time of wheel acceleration sign change, a difference caused by the differences in error signal.

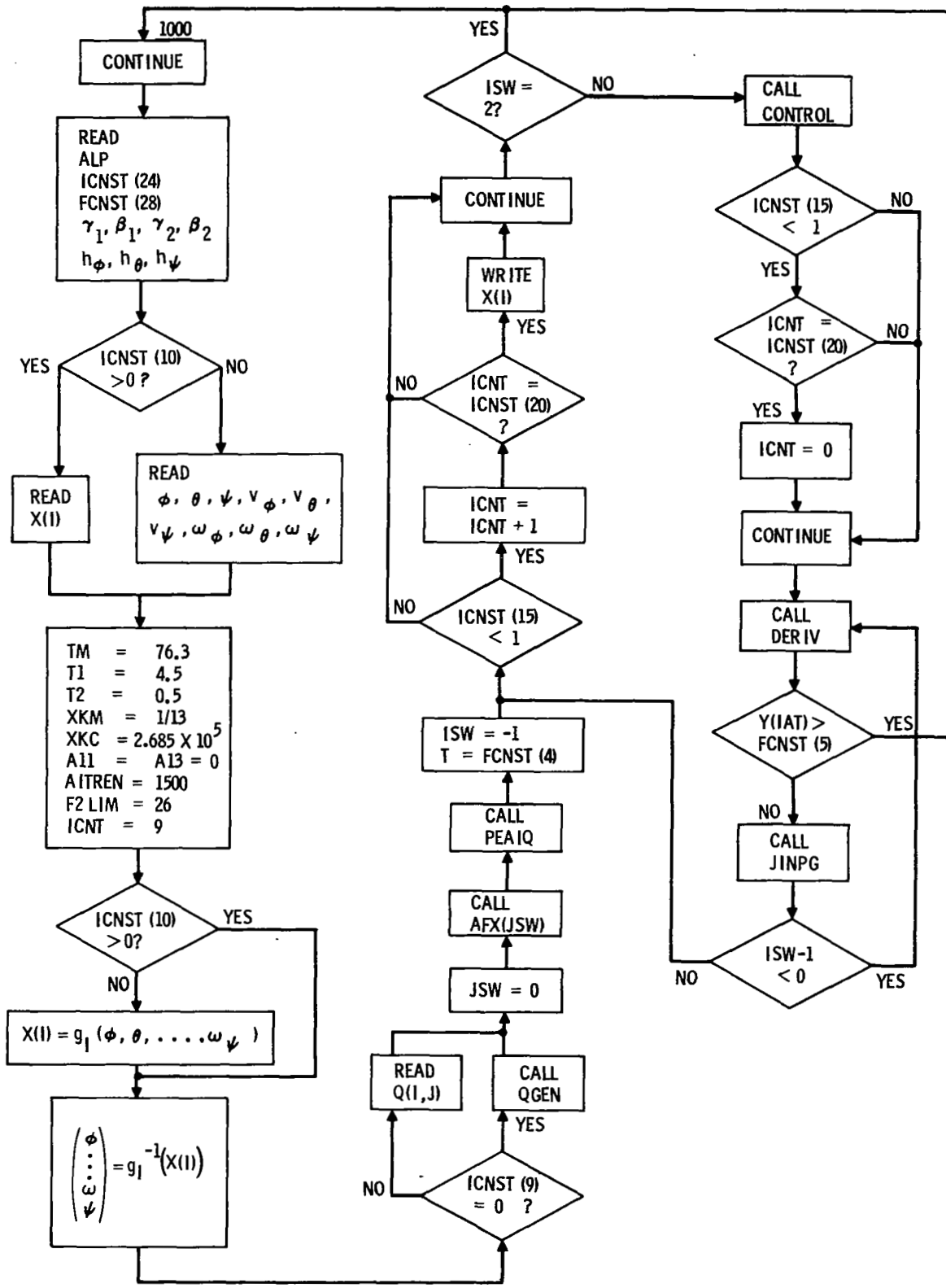


Fig. B-1 Generic Simulation Program Flow Chart

Run #	$\sin(\gamma_{1c} - \gamma_{2c})$	Inner Gimbal Commands Degrees β_{1c}	β_{2c}	Total Axis Momenta ft-lb sec Ih^o_ϕ	Ih^o_θ	Ih^o_ψ
1	0.1	0.0	-30.0	0.0	0.0	0.0
2	0.1	30.0	-30.0	0.0	0.0	0.0
3	0.1	30.0	-30.0	1.0	1.0	1.0
4	1.0	0.0	-30.0	0.0	0.0	0.0
5	1.0	0.0	-30.0	1.0	1.0	1.0

Initial Conditions: $\phi, \theta, \psi = 15.0^\bullet$; $v_\phi, v_\theta, v_\psi = 1$ ft-lb$_f$ sec; $\omega_\phi, \omega_\theta, \omega_\psi = 100$ volts

System Parameters: $\tau_1 = 4.5$ sec; $\tau_2 = 0.5$ sec; $K_c = 2.685 \cdot 10^5$ volt/rad; $K_m = 1/13$ ft-lb$_f$-sec/volt; $I = 1500$ slug-ft^2

Basic Nonlinear Model	- AN	Motor Voltage Only Nonlinearity	- MV
Basic Model with Limiting	- ANL	Six Dimensional Model	- 6D

Fig. B-2 Control System Simulation Run Plan

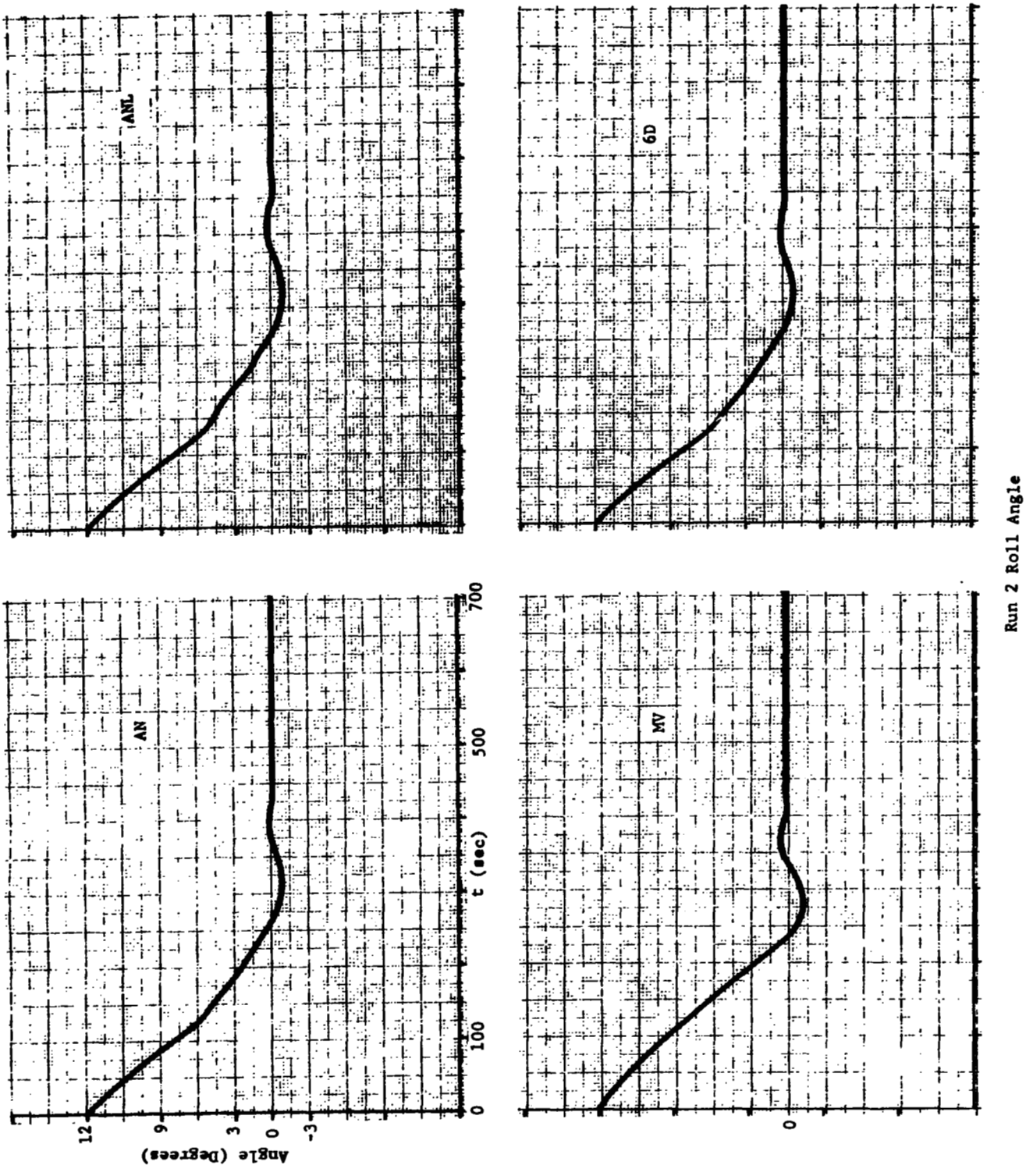


Fig. B-3 Control System Simulation Run 2 (Sheet 1 of 6)

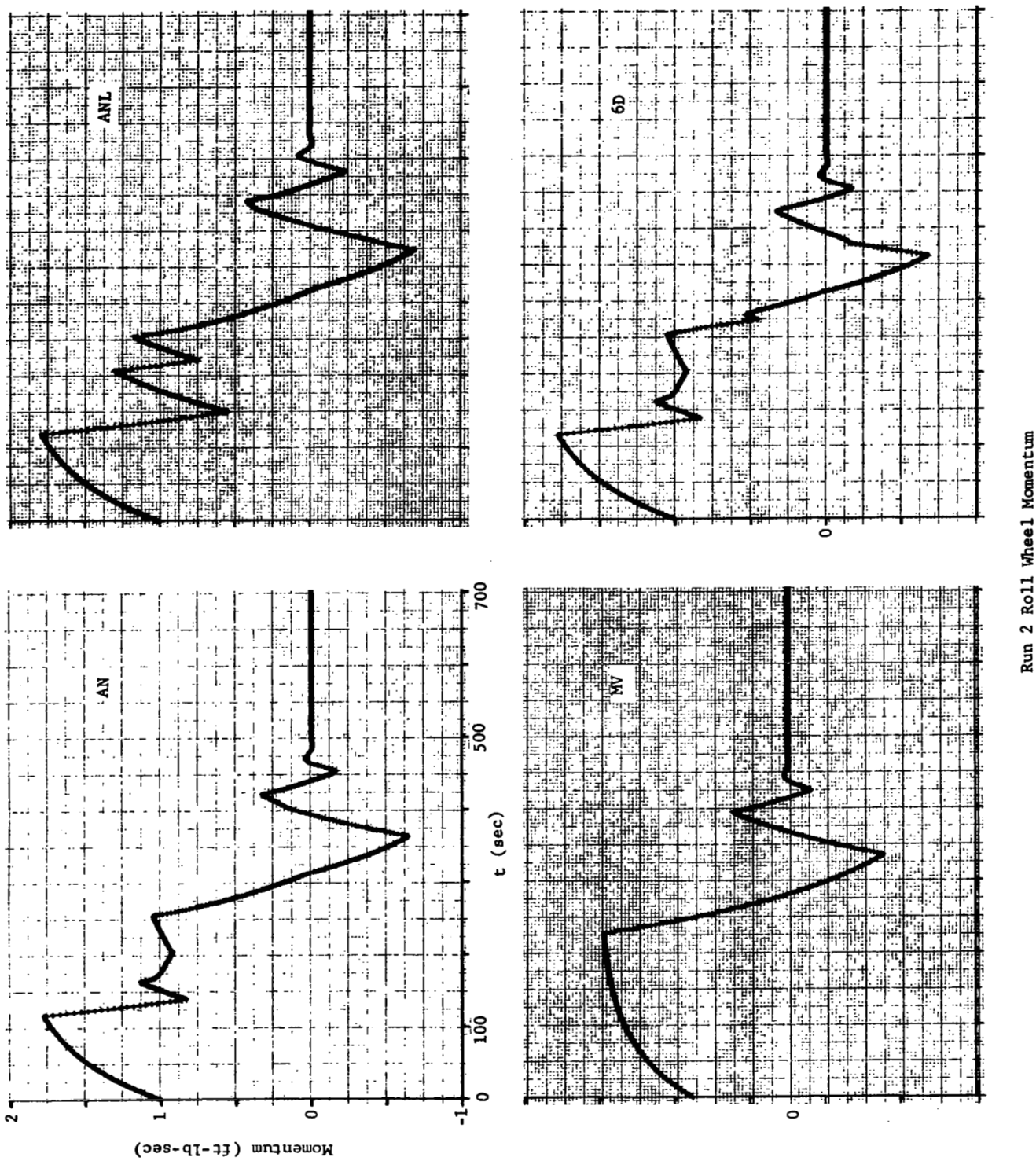


Fig. B-3 Control System Simulation Run 2 (Sheet 2 of 6)

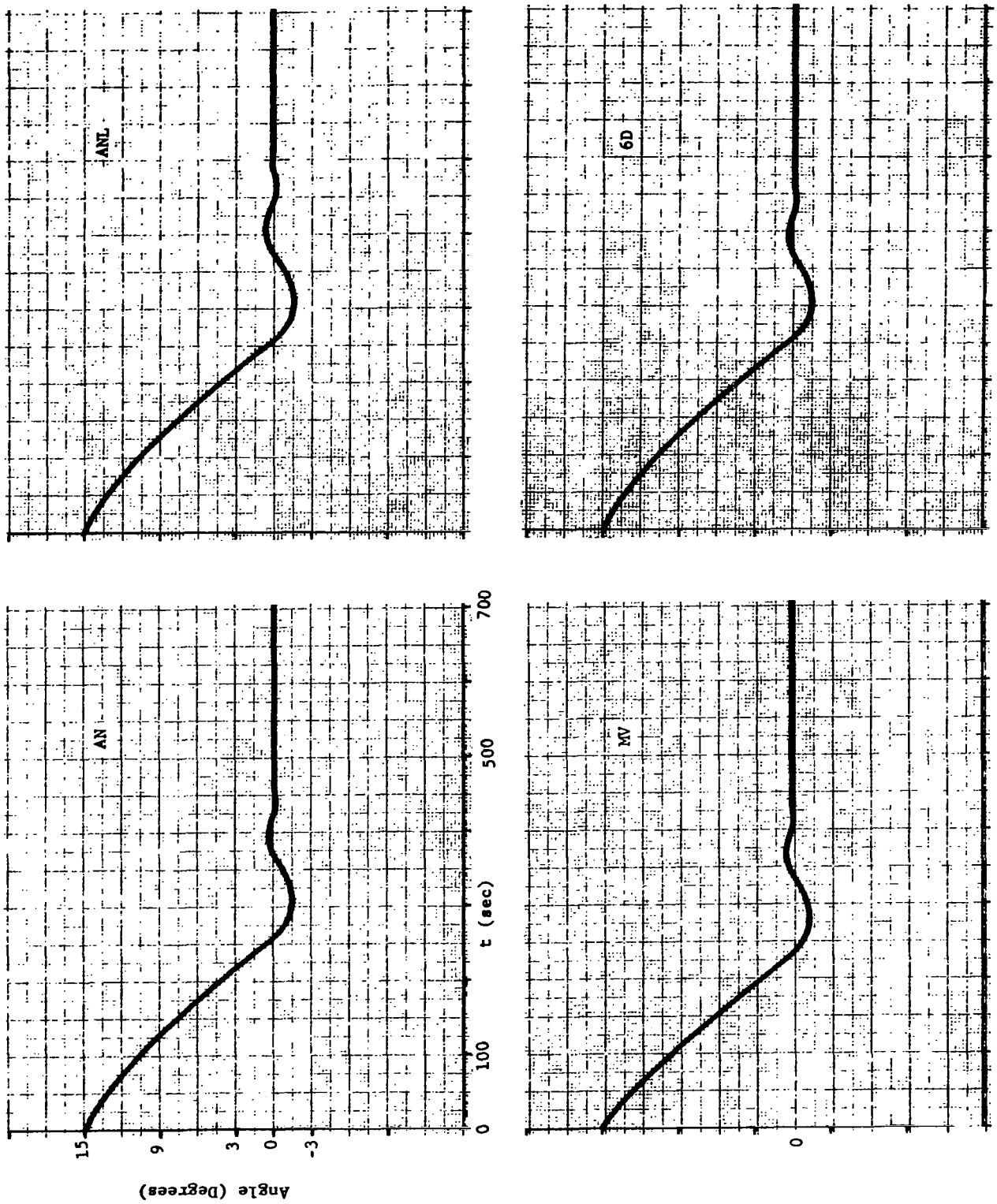


Fig. B-3 Control System Simulation Run 2 (Sheet 3 of 6)

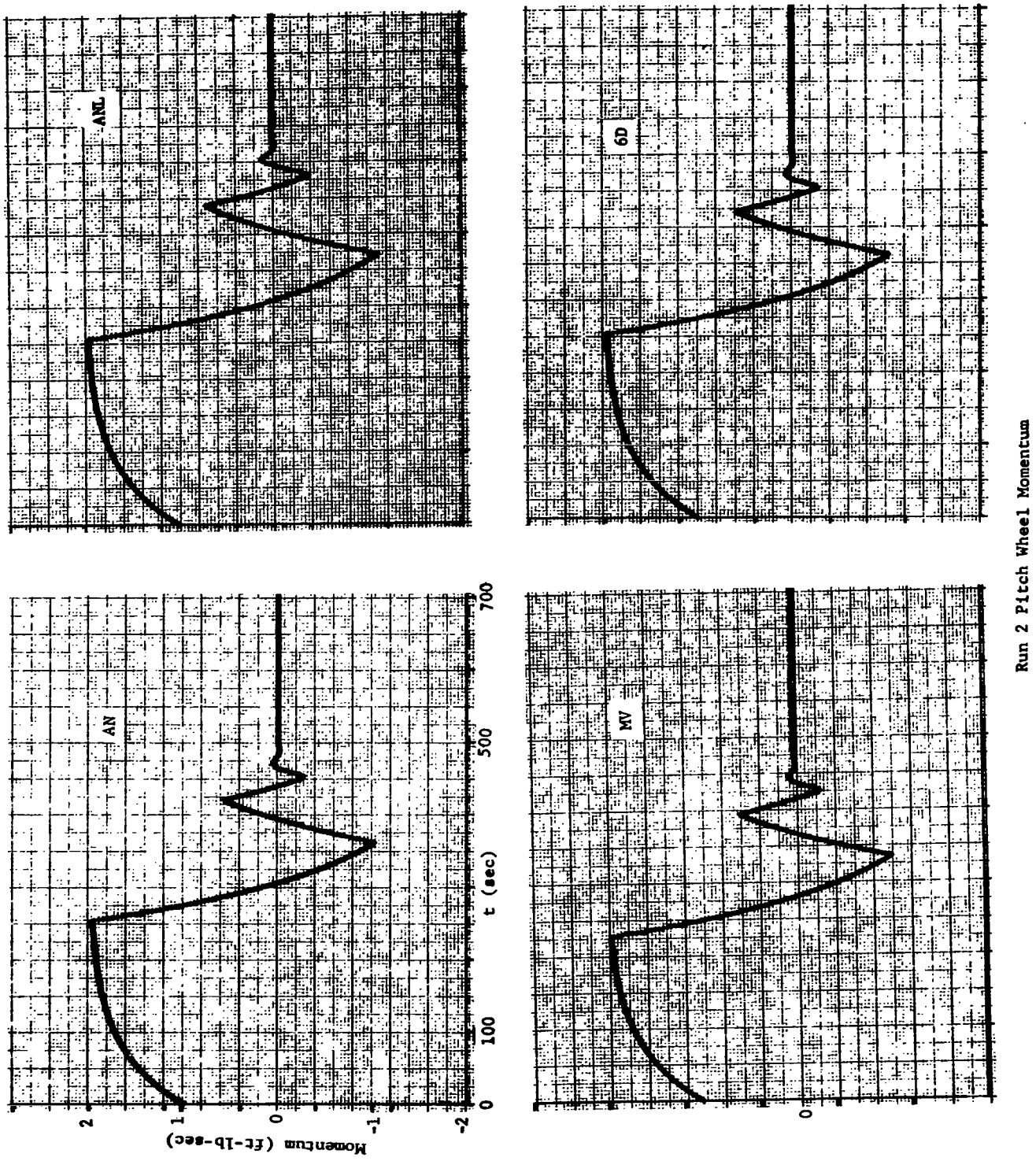


Fig. B-3 Control System Simulation Run 2 (Sheet 4 of 6)

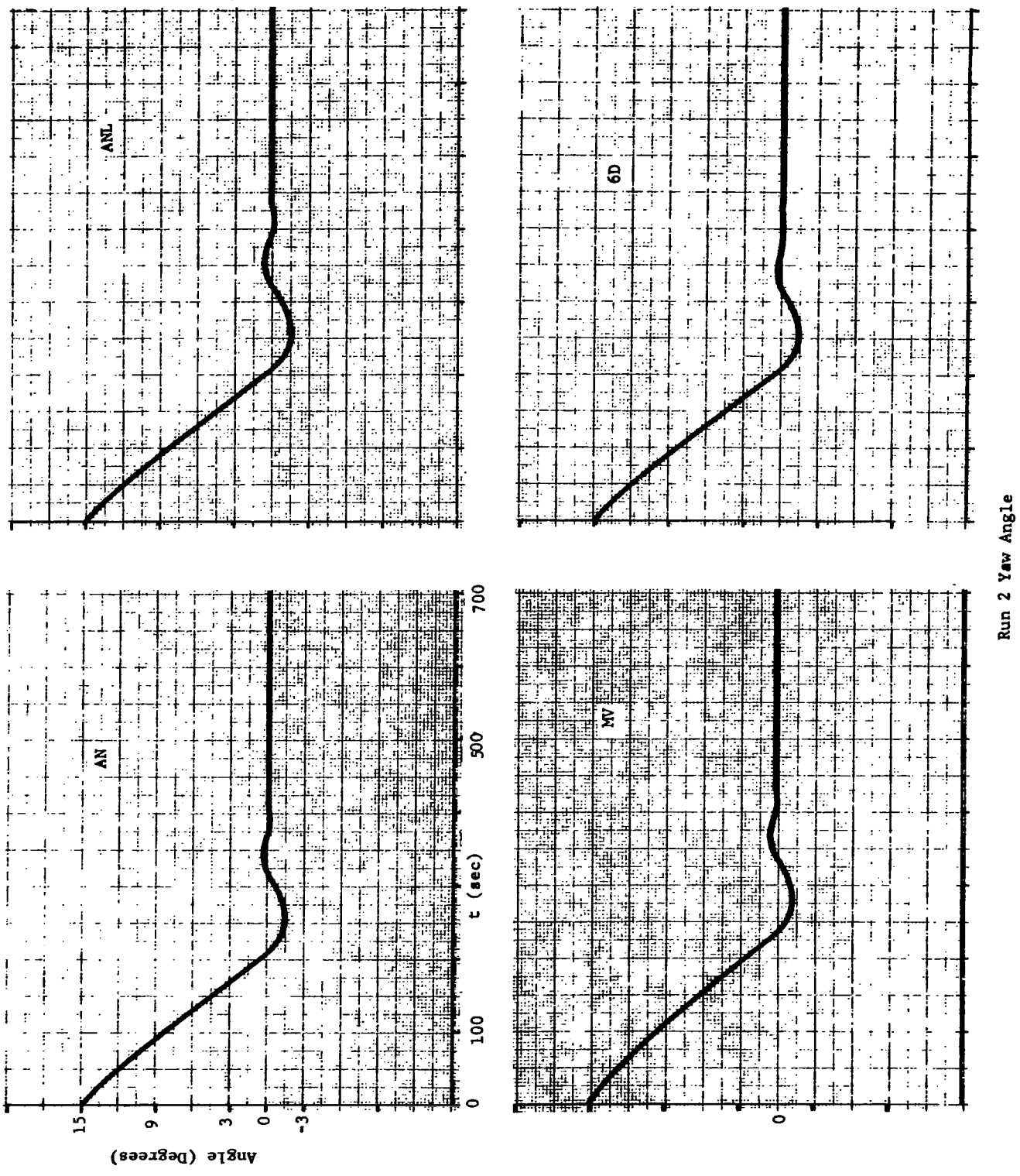


Fig. B-3 Control System Simulation Run 2 (Sheet 5 of 6)

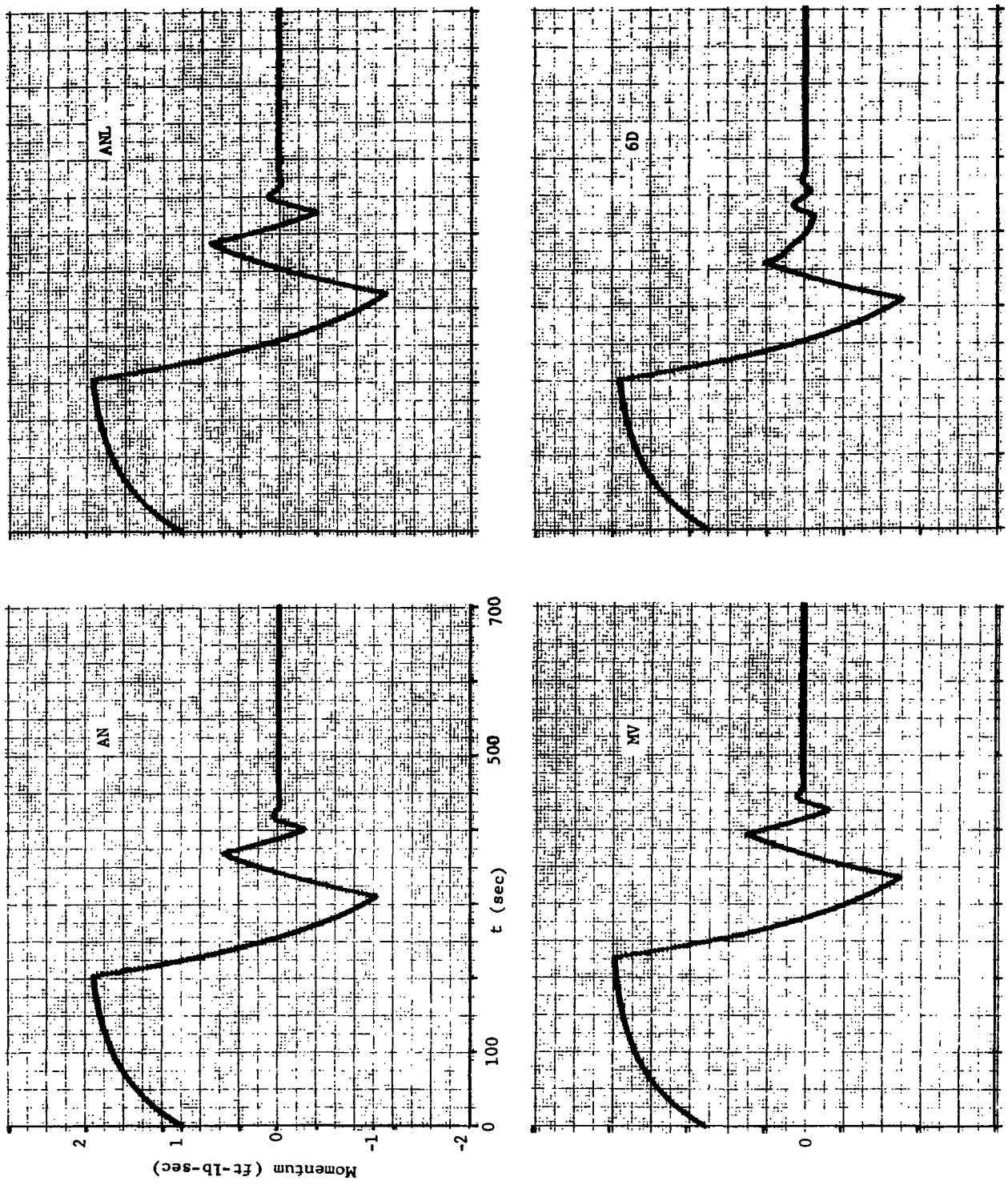


Fig. B-3 Control System Simulation Run 2 (Sheet 6 of 6)

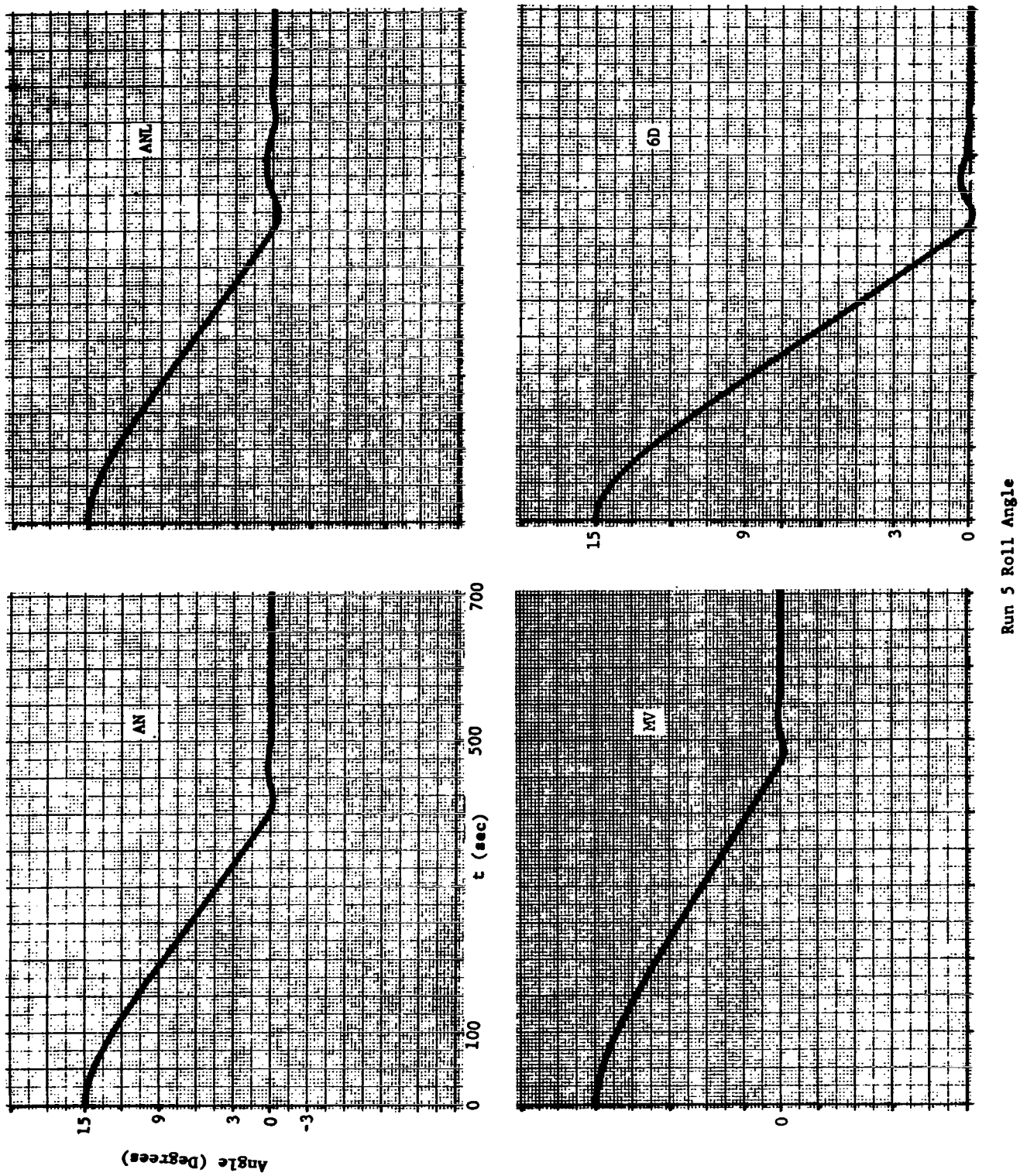


Fig. B-4 Control System Simulation Run 5 (Sheet 1 of 6)

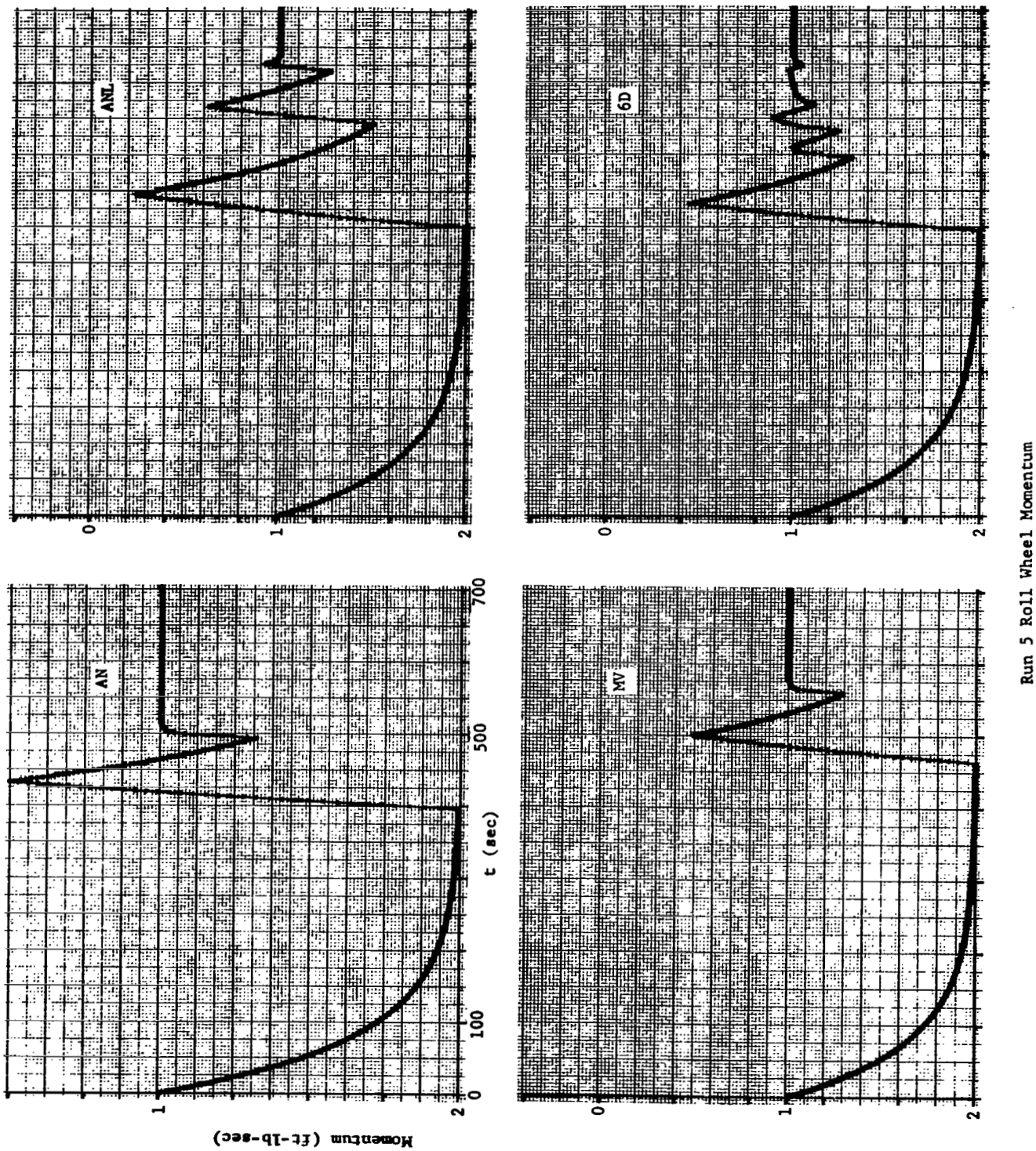


Fig. B-4 Control System Simulation Run 5 (Sheet 2 of 6)

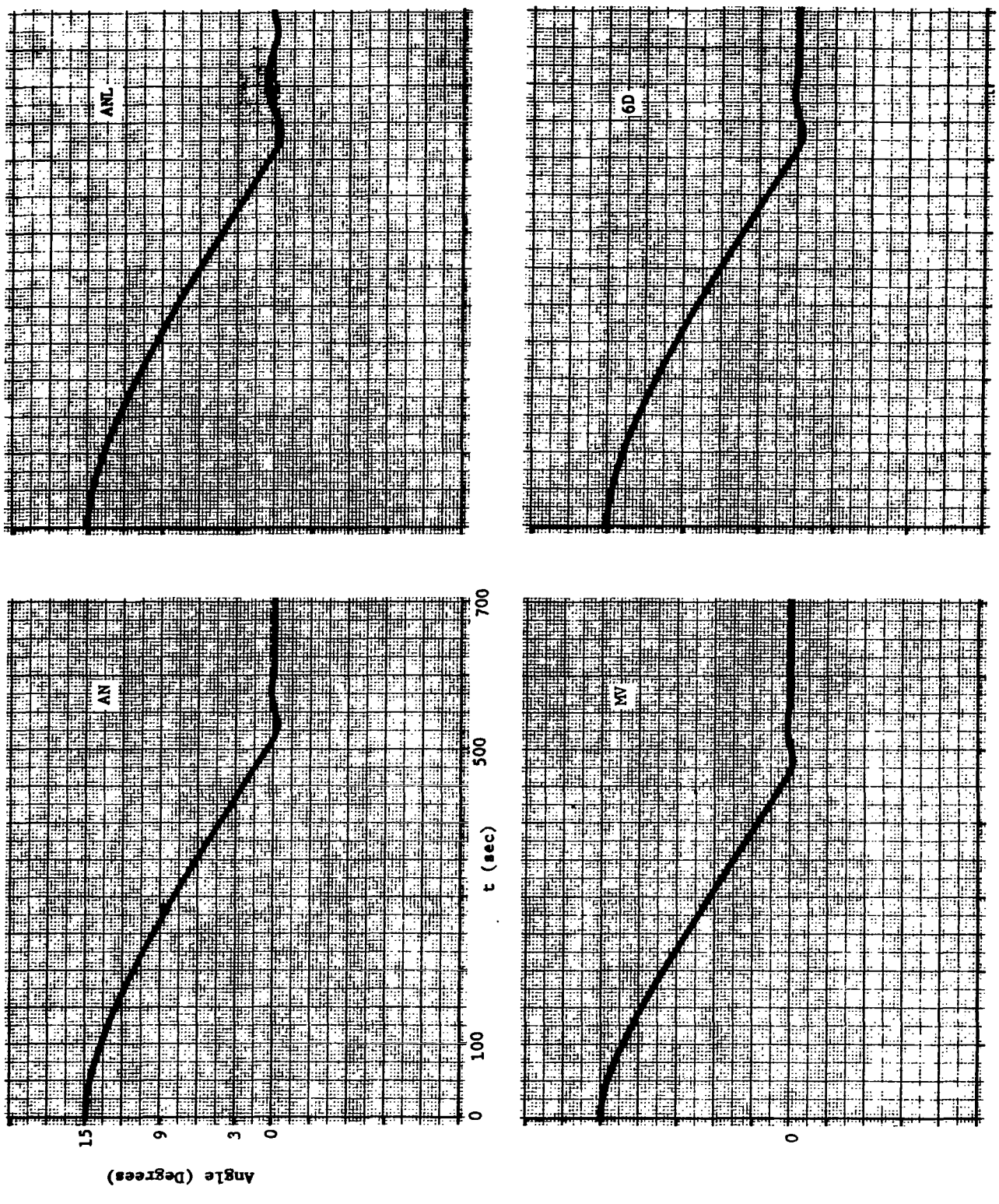


Fig. B-4 Control System Simulation Run 5 (Sheet 3 of 6)

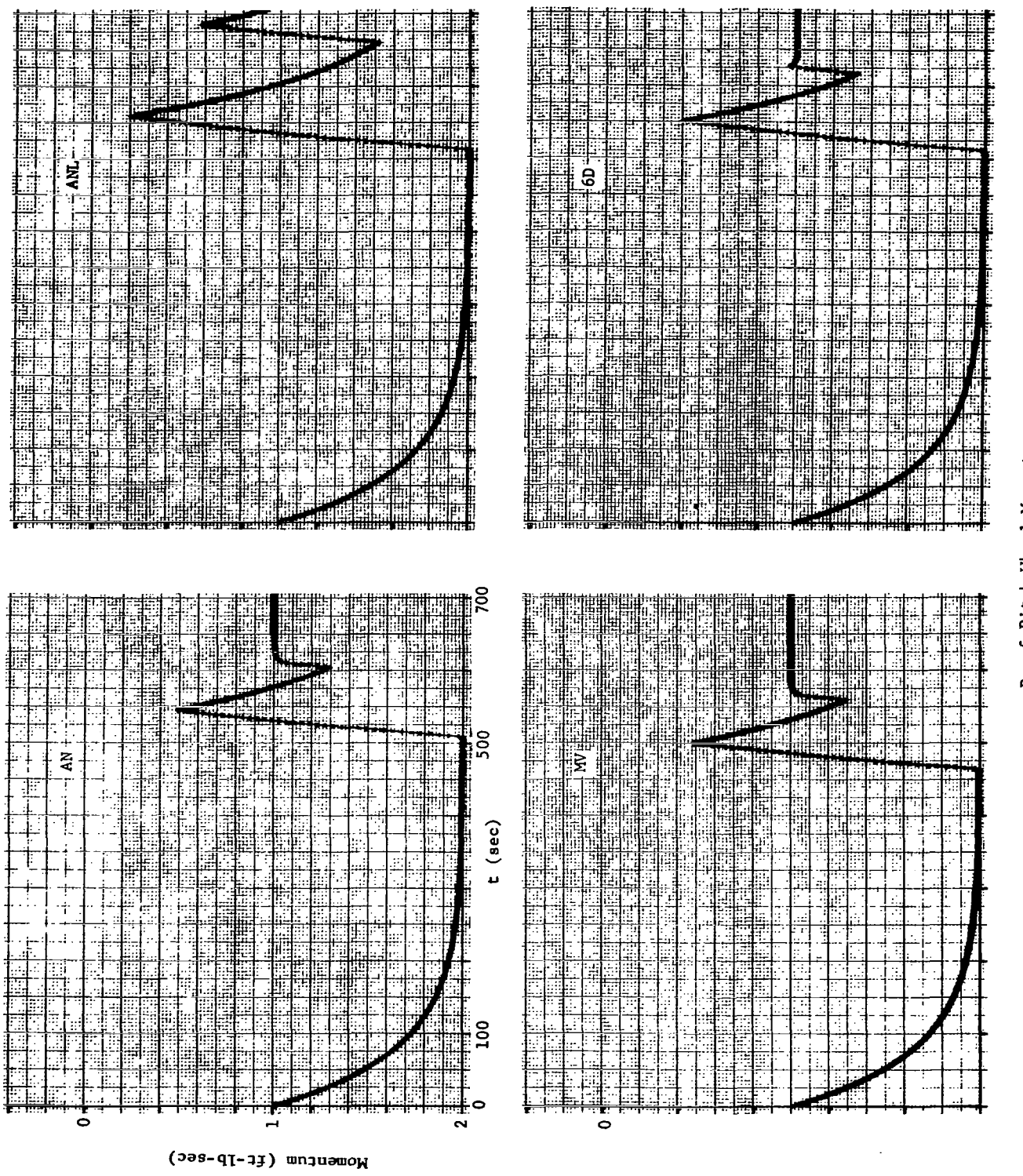


Fig. B-4 Control System Simulation Run 5 (Sheet 4 of 6)

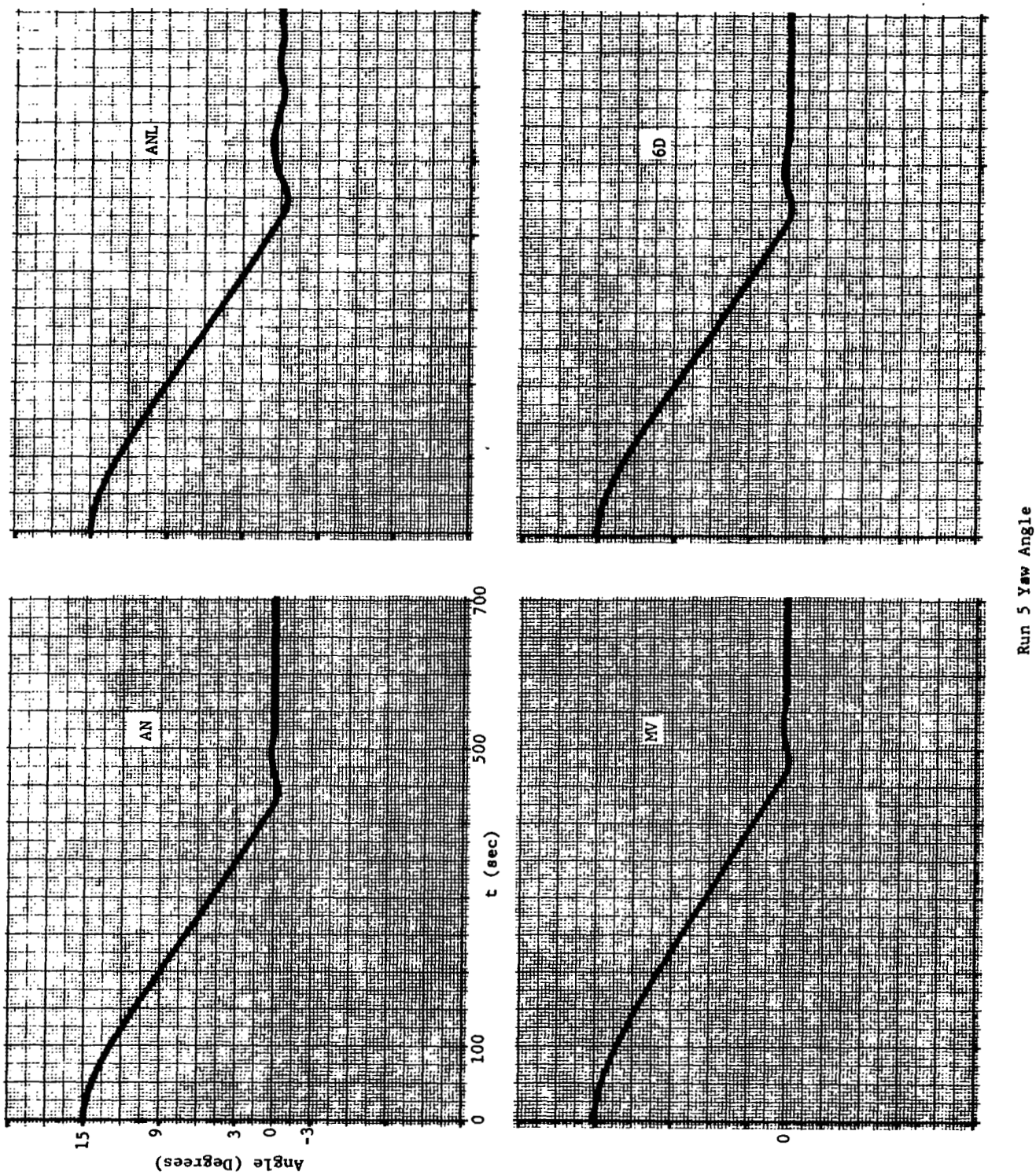


Fig. B-4 Control System Simulation Run 5 (Sheet 5 of 6)

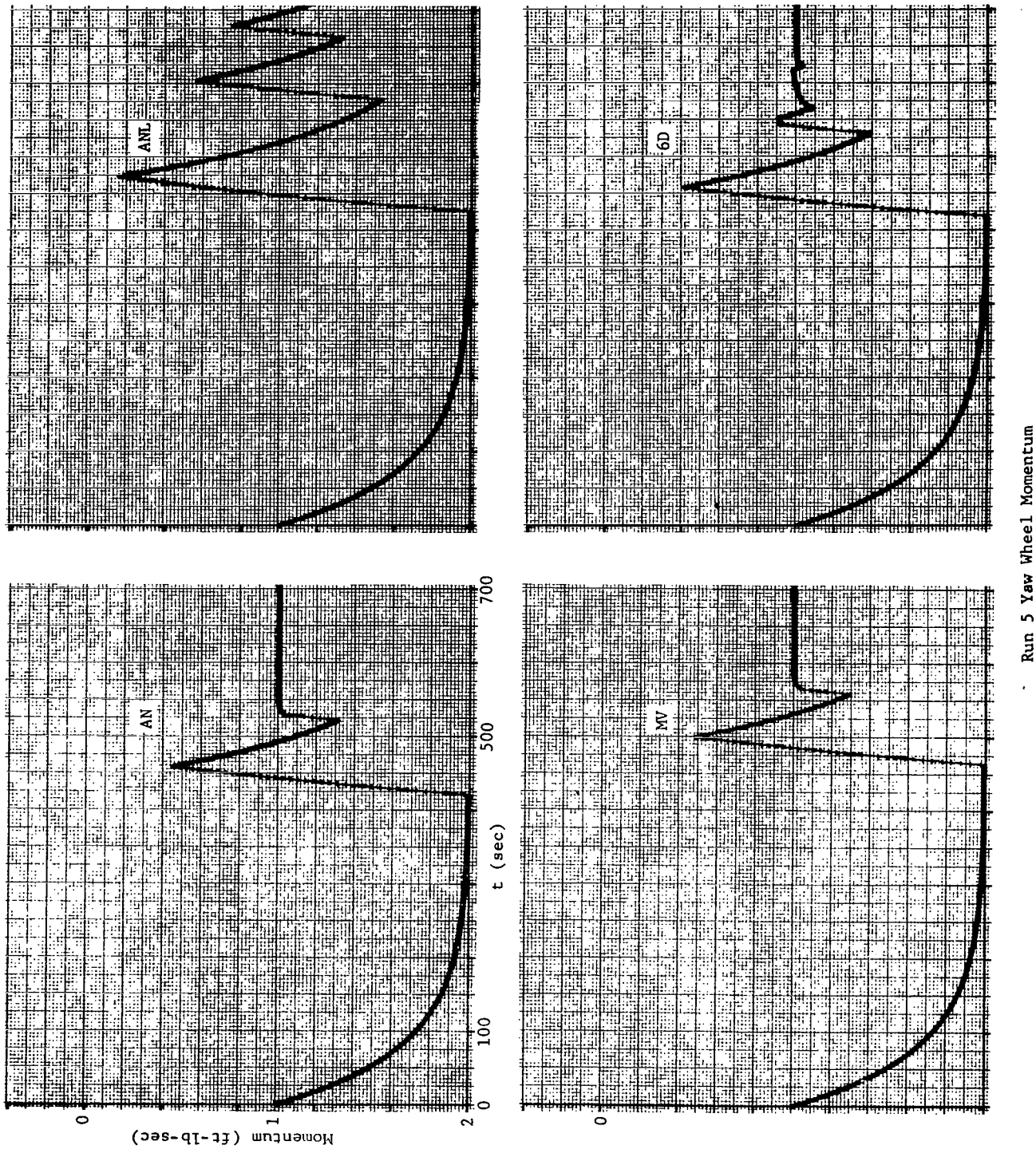


Fig. B-4 Control System Simulation Run 5 (Sheet 6 of 6)

A run was then made with initial condition identical to those in Fig. B-2, but with $s(\gamma_{1c} - \gamma_{2c}) = 0.1$ and $\beta_{1c} = \beta_{2c} = 30$. The results of this run are presented in Fig. B-5. It can be seen that "AN," "6-D," "ANL" are unstable and that "MV" is stable. Study of this case reveals the fact that at $t = 0$ the error signals ϵ_ϕ, ϵ_θ, ϵ_ψ were negative for "AN," "ANL," "6-D" but positive (the necessary sign for stability for initial conditions chosen) for "MV." Thus the nonlinear coupling in the true error signals can provide the wrong error voltage sign for some tracker cases. Table B-1 presents one set of tracker commands which will yield this situation for $\phi = \theta = \psi = 10°$ at $t = 0$. In all cases the restriction $|\gamma_{1c} - \gamma_{2c}| \geq 10°$ is obeyed. These cases should all be unstable for "AN," "ANL," and "6-D" but stable for "MV."

From this information it can be concluded that in general neither "MV" nor any other model that is based on linearizing the feedback is valid for a stability study. The exact model, "AN," must be used.

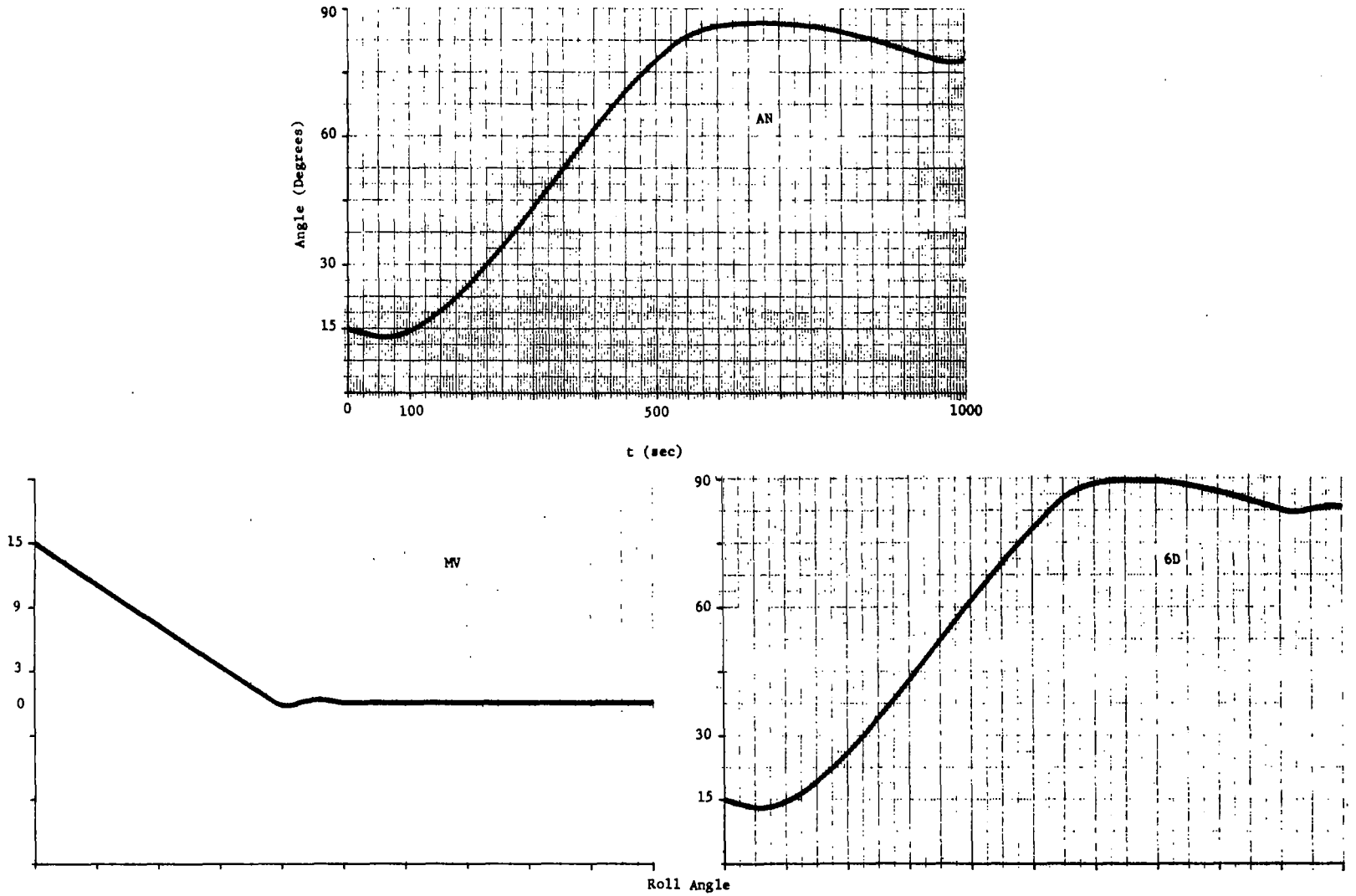
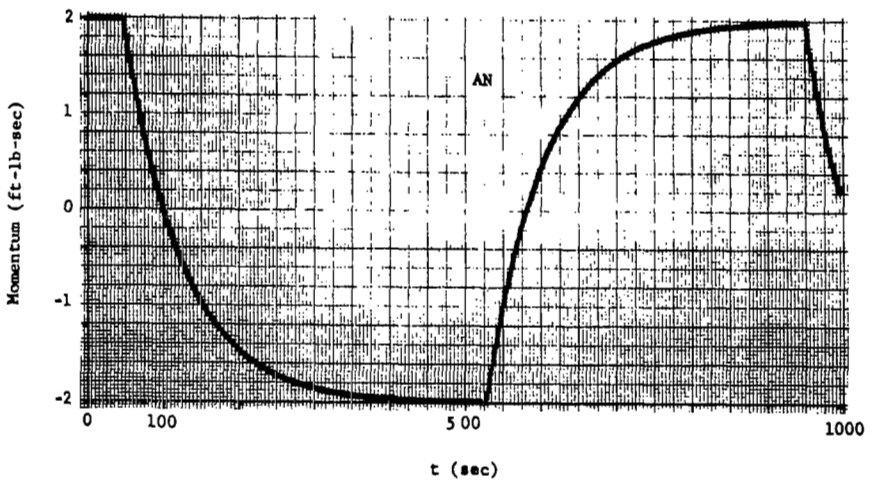


Fig. B-5 Control System Simulation Unstable Case (Sheet 1 of 6)



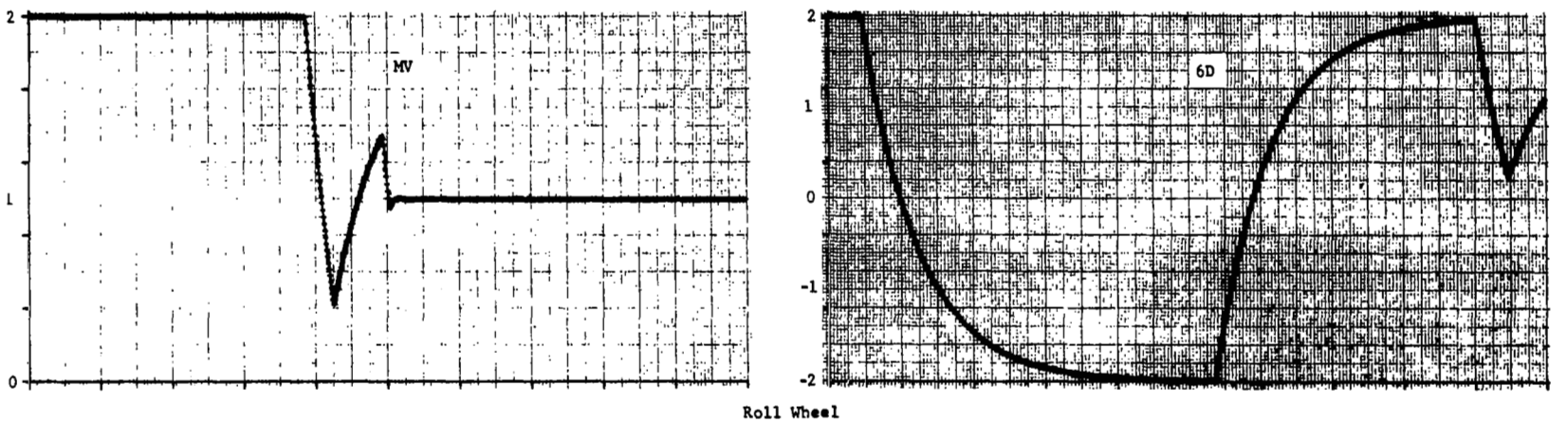
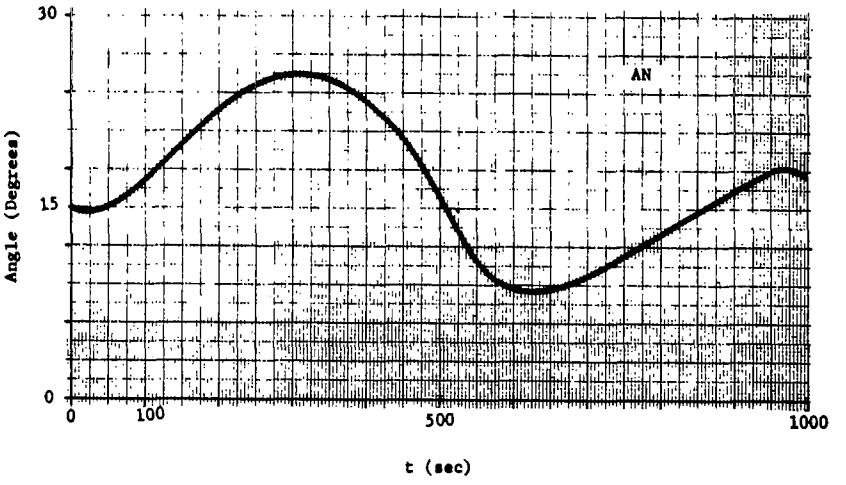


Fig. B-5 Control System Simulation Unstable Case (Sheet 2 of 6)



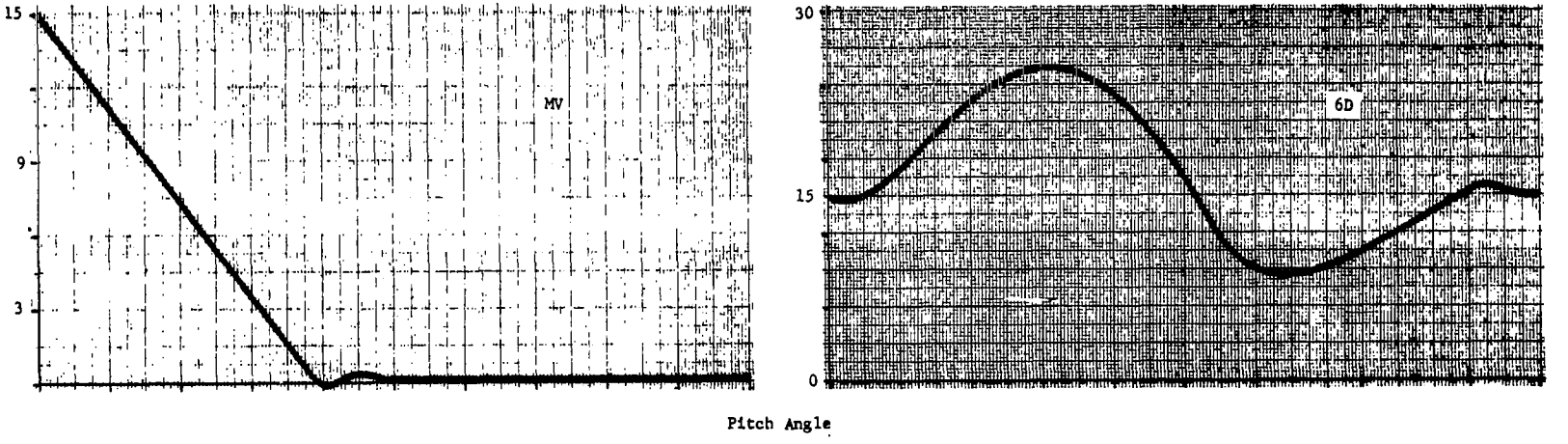


Fig. B-5 Control System Simulation Unstable Case (Sheet 3 of 6)

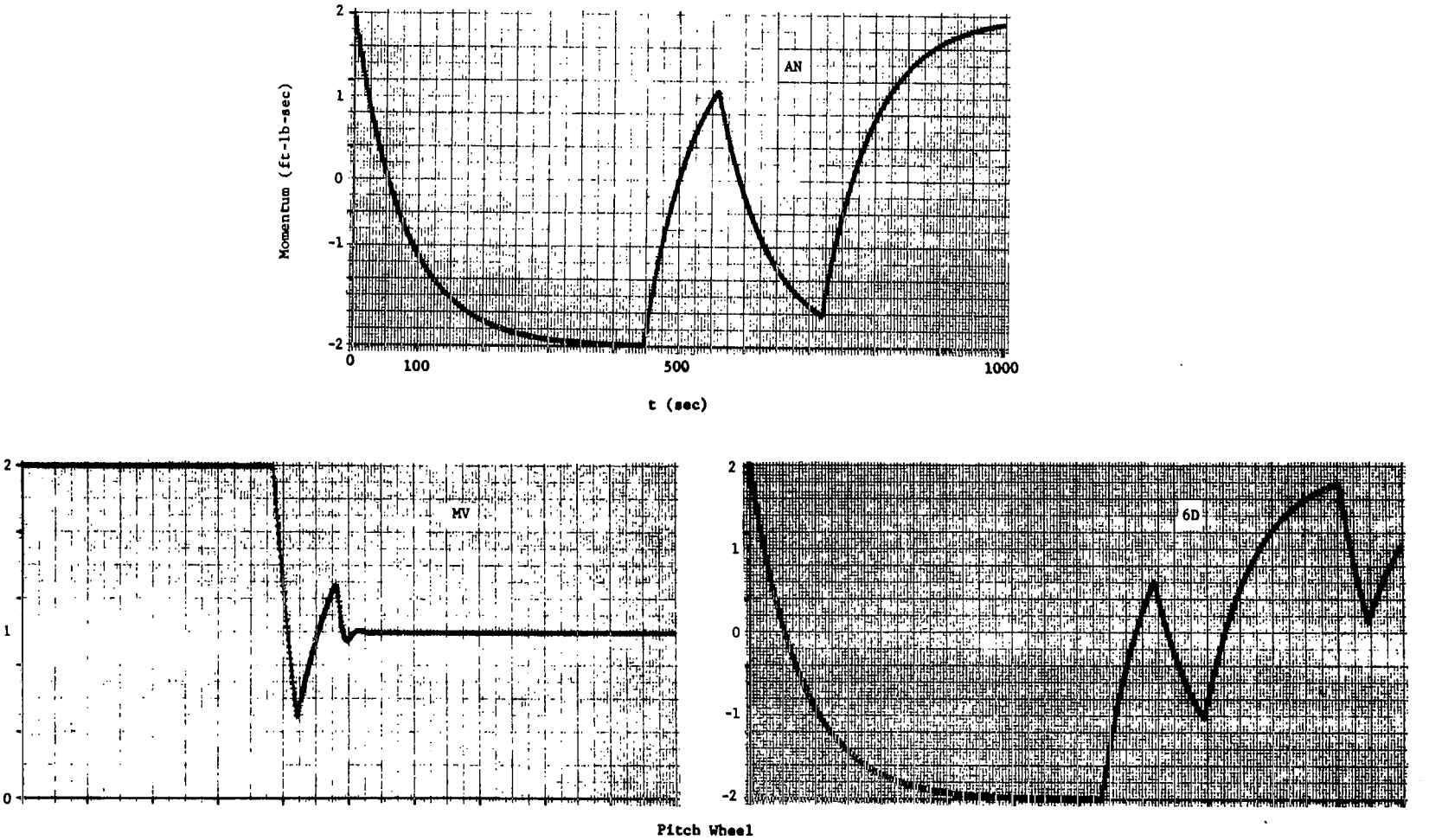


Fig. B-5 Control System Simulation Unstable Case (Sheet 4 of 6)

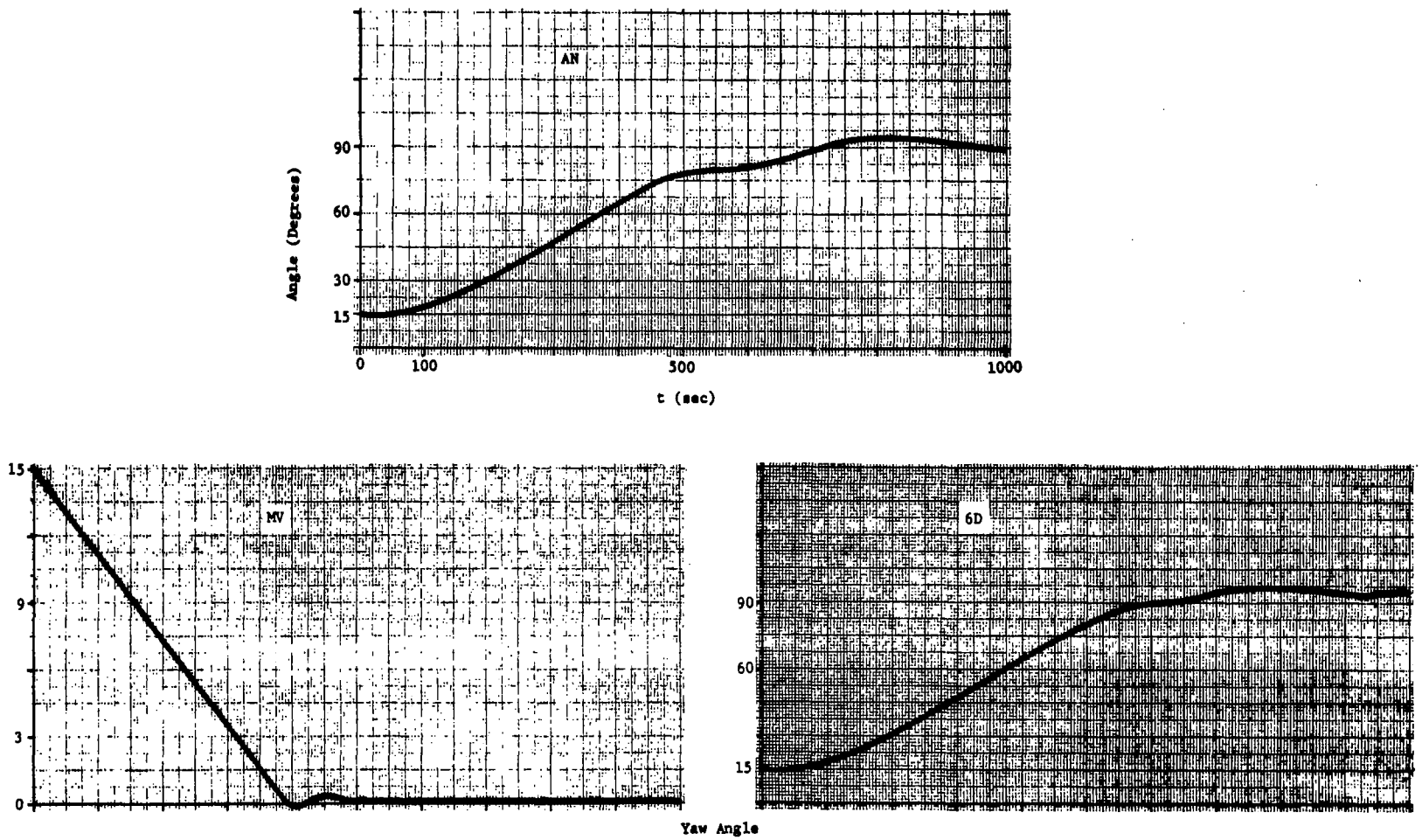


Fig. B-5 Control System Simulation Unstable Case (Sheet 5 of 6)

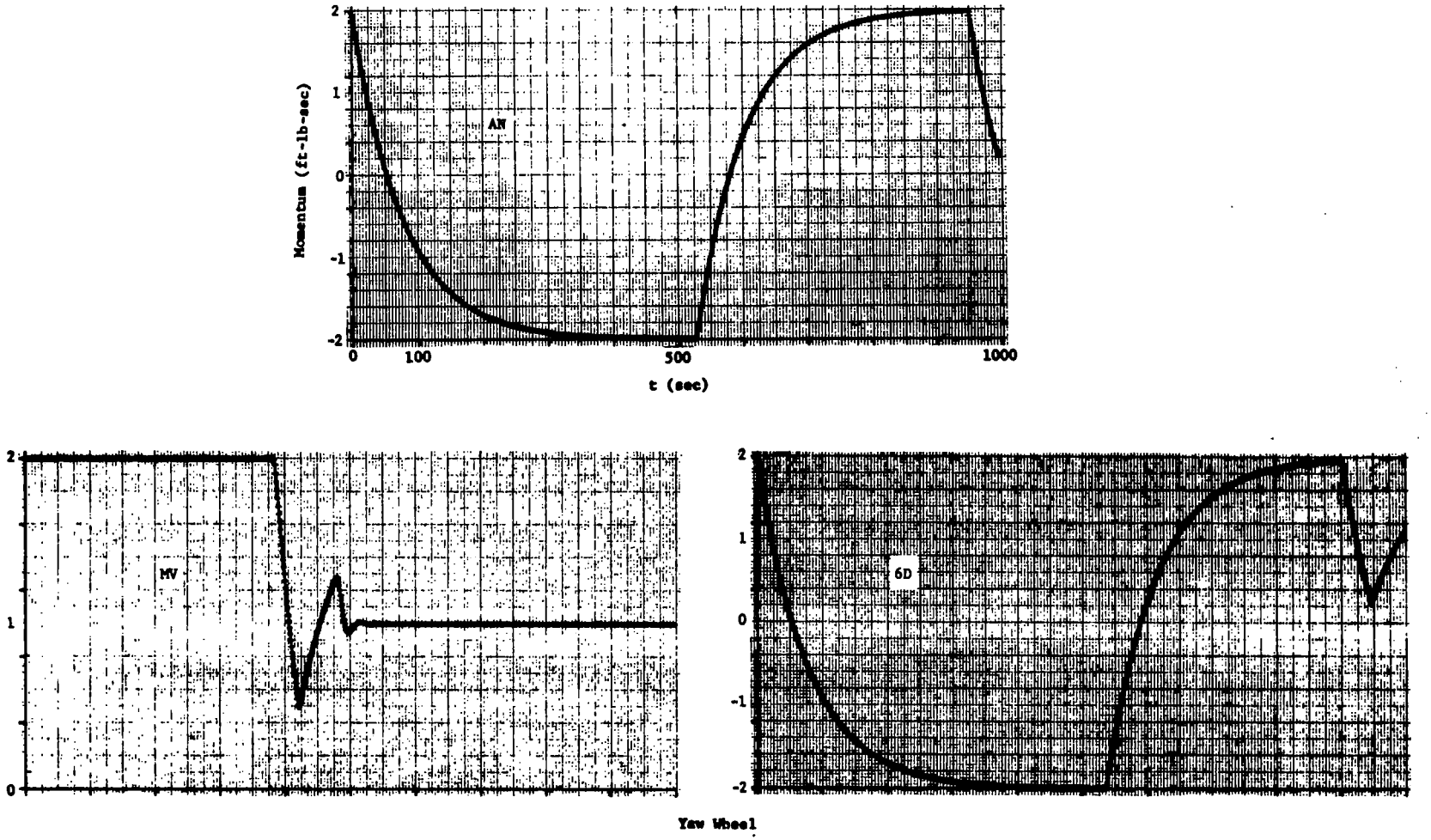


Fig. B-5 Control System Simulation Unstable Case (Sheet 6 of 6)

Table B-1

PRESUMED UNSTABLE CASES

Command Gimbal Angles (Degrees)

γ_{1c}	γ_{2c}	β_{1c}	β_{2c}*
5	-5	40	50
5	-5	45	40
5	-5	50	35
5	-5	55	25
5	-5	60	10
10	0	45	45
10	0	50	35
10	0	55	25
10	0	60	5

at $t = 0$ $\Phi = \theta = \psi = 10°$; $\omega_\Phi = \omega_\theta = \omega_\psi = 0$; $h_\Phi^o = h_\theta^o = h_\psi^o = 0$

*This value and higher lead to predicted unstable cases for these values at $t = 0$

$$e_x = e_y = e_z < 0$$

APPENDIX C

ANALYTICAL STABILITY STUDIES

In this appendix the stability analysis of the linear part of the system described in Section 4 is shown to be quite straightforward as a result of the linear decoupling feature of the "paired-tracker" design. In addition, the stability of the Popov approximation (motor saturation only) is examined and it is shown that, at present, the available frequency domain techniques are not sufficiently well developed to completely prove the absolute stability of that model. Note that in Appendix B it was shown that this approximation does not adequately represent the stability properties of the actual system; however, an examination of its stability properties is required in Appendix F, part (ii).

If all nonlinearities of the system are ignored, then the system equations reduce to

$$\dot{x} = Ax \tag{C-1}$$

where x is a nine vector as before and A can be partitioned into 3×3 submatrices, i.e.,

$$A = \begin{bmatrix} A_1 & A_5 & A_6 \\ 0 & A_2 & 0 \\ 0 & 0 & A_3 \end{bmatrix} \tag{C-2}$$

where the A_i are readily obtained from Fig. 4-6 of Section 4.

The stability of the linear system (C-1) is determined by the roots of the characteristic equation

$$c(\lambda) = \det(\lambda I - A) = 0 \ . \tag{C-3}$$

Since A is upper triangular, in partitioned form, we obtain

$$c(\lambda) = \det(\lambda I_3 - A_1)\det(\lambda I_3 - A_2)\det(\lambda I_3 - A_3) = 0 \tag{C-4}$$

where I_3 is the 3×3 identity matrix. Thus, this linear system is stable if and only if each channel of the linear system is stable. This is a direct result of the linear decoupling inherent in the "paired-tracker" design.

Thus, the stability analysis of the linear system is reduced to determining the stability of each channel separately, i.e., determining the stability of each matrix A_i for $i = 1, 2, 3$. The problem is further simplified by recognizing that $A_2 = A_3$. Stability of the system (C-1) is obtained if the characteristic roots of A_1 and A_2 have negative real parts. Their corresponding characteristic matrices are:

$$(\lambda I_3 - A_1) = \begin{bmatrix} \lambda & \frac{K_m K_c}{I} & 0 \\ -\frac{\tau_1 + \tau_2}{\tau_m \tau_2} & \left(\lambda + \frac{1}{\tau_m}\right) & -\frac{\tau_1}{\tau_m \tau_2} \\ \frac{1}{\tau_2} & 0 & \left(\lambda + \frac{1}{\tau_2}\right) \end{bmatrix} \qquad \text{(C-5)}$$

and

$$(\lambda I_3 - A_2) = \begin{bmatrix} \lambda & \frac{K_m K_c}{I} & 0 \\ -a_1 \frac{\tau_1 + \tau_2}{\tau_m \tau_2} & \left(\lambda + \frac{1}{\tau_m}\right) & -\frac{\tau_1}{\tau_m \tau_2} \\ \frac{a_1}{\tau_2} & 0 & \left(\lambda + \frac{1}{\tau_2}\right) \end{bmatrix} \qquad \text{(C-6)}$$

where $a_1 = d_{12} s(\gamma_{1c} - \gamma_{2c})$. The characteristic equations are

$$c_1(\lambda) = \det(\lambda I_3 - A_1) = \lambda\left(\lambda + \frac{1}{\tau_m}\right)\left(\lambda + \frac{1}{\tau_2}\right) + \frac{K_m K_c}{I} \frac{\tau_1 + \tau_2}{\tau_m \tau_2}\left(\lambda + \frac{1}{\tau_1 + \tau_2}\right) = 0 \qquad \text{(C-7)}$$

and

$$c_2(\lambda) = \det(\lambda I_3 - A_2) = \lambda\left(\lambda+\frac{1}{\tau_m}\right)\left(\lambda+\frac{1}{\tau_2}\right) + a_1 \frac{K_m K_c}{I} \frac{\tau_1+\tau_2}{\tau_m \tau_2}\left(\lambda+\frac{1}{\tau_1+\tau_2}\right) = 0 \tag{C-8}$$

The roots of these equations can be found by root locus techniques from the equation

$$1 + a_1 \frac{K_m K_c}{I} \frac{\tau_1+\tau_2}{\tau_m \tau_2} \frac{\left(\lambda+\frac{1}{\tau_1+\tau_2}\right)}{\lambda\left(\lambda+\frac{1}{\tau_m}\right)\left(\lambda+\frac{1}{\tau_2}\right)} = 0 \tag{C-9}$$

where $a_1 = 1$ for the roots of $c_1(\lambda) = 0$ and $a_1 = d_{12}s(\gamma_{1c} - \gamma_{2c})$ for the roots of $c_2(\lambda) = 0$. Substitution of the parameter values yields

$$1 + a_1(1.79) \frac{\lambda + 0.2}{\lambda(\lambda + 0.013)(\lambda + 5)} = 0 \tag{C-10}$$

Simple root locus considerations show that the roots of (C-10) have negative real parts for all $a_1 > 0$, in fact, the roots of (C-9) have negative real parts for all positive finite values of the parameters. Thus, the linear system (C-1) is stable for all positive values of the system parameters. Note that for a_1 to be positive it is necessary that $\text{sgn } d_{12} = \text{sgn}(\gamma_{1c} - \gamma_{2c})$. This is how the functional form of the choice $d_{12} = 2.0\ \text{sgn}(\gamma_{1c} - \gamma_{2c})$ is arrived at.

Thus, we have shown that the stability analysis of the linear part of the system can be accomplished by direct application of simple root locus concepts, and that this is a result of the linear decoupling feature of the "paired-tracker" design.

The stability analysis of the Popov approximation to the system model should be equally straightforward; it is not, however, because the available frequency domain techniques are not fully developed. This approximate model was derived in Appendix A and is summarized by

$$u = - W(s) f^a(u) \tag{C-11}$$

where

$$W(s) = w(s)W$$

$$w(s) = \frac{K_m K_c}{I} \left[\frac{(\tau_1 + \tau_2)s + 1}{s(\tau_m s + 1)(\tau_2 s + 1)} \right] \tag{C-12}$$

$$W = \begin{bmatrix} 1 & -t\beta_{1c} c\gamma_{1c} & t\beta_{1c} s\gamma_{1c} \\ 0 & d_{12} s(\gamma_{1c} - \gamma_{2c}) & 0 \\ 0 & 0 & d_{12} s(\gamma_{1c} - \gamma_{2c}) \end{bmatrix}$$

$$f_i^a(u) = 26 \; \text{sat} \left[\frac{1}{26} \left(u_i + \frac{Ih_i^o}{K_m} \right) \right] - \frac{Ih_i^o}{K_m} , \qquad i = 1, 2, 3$$

and u is a three vector.

Let us first consider the special case $\beta_{1c} = 0$ in which event the three channels are completely uncoupled. Then we apply the special form of the Popov theorem [15] to a single channel.

Theorem (Popov). — For the particular case of a system (the linear part has poles with zero real part) to be absolutely stable in the sector $[\epsilon, K]$, $\epsilon > 0$ (i.e., $\epsilon u_i^2 \leq u_i f_i^a(u_i) \leq K u_i^2$, for all $u_i \neq 0$), it is sufficient that there exists a real finite number q such that for all ω

$$\text{Re}(1 + i\omega q) W(i\omega) + \frac{1}{K} > 0 \tag{C-13}$$

and that the condition for stability in the limit (i.e., if there is a single pole at the origin then $\lim_{\omega \to 0^+} \text{Im } W(i\omega) = -\infty$) is satisfied.

In this uncoupled case we have, for a single channel

$$W(i\omega) = a_1 \frac{K_m K_c}{I}\left[\frac{(\tau_1 + \tau_2)i\omega + 1}{i\omega(\tau_m i\omega + 1)(\tau_2 i\omega + 1)}\right] \qquad \text{(C-14)}$$

where a_1 is either 1 or $d_{12}s(\gamma_{1c} - \gamma_{2c})$, and $(1 + i\omega q)W(i\omega)$ is positive real since the poles and zeros of $W(i\omega)$ are all real, the gain is positive, and q can be chosen to make the poles and zeros of $(1 + i\omega q)W(i\omega)$ interlace. In addition, the condition for stability in the limit holds since the gain is positive. Unfortunately, the saturation function does not fall into the $[\epsilon, K]$ sector since for large $|u_i|$ the gain of $f_i^a(u_i)$ goes to zero. Perhaps one can try to use mathematical artifices to make the system satisfy the theorem but no motor known to us can develop unlimited speed for unlimited input for any finite time. It is the pole at the origin that causes the failure of this application of the Popov theorem.

In [16] Brockett quotes another version of the Popov theorem, viz.,

> Theorem (Brockett). — Let $q(s)$ and $p(s)$ be polynomials without common factors and let
>
> $$\dot{x} = Ax + bu \; ; \; y = c^T x \qquad \text{(C-15)}$$
>
> be an irreducible (controllable and observable) representation of $G(s) = q(s)/p(s)$. Suppose $p(s)$ has no zeros in the half-plane $\mathrm{Re}\, s > 0$. It follows that the nonlinear system
>
> $$\dot{x} = Ax - bf(c^T x) \; ; \; 0 \leq f(y)/y \leq K \qquad \text{(C-16)}$$
>
> has a null solution which is asymptotically stable in the large provided there exists a real q such that $(1 + qs)G(s) + 1/K$ is positive real and
>
> $$f^a(y)\left(Ky - f^a(y)\right) \neq 0 \quad \text{for} \quad y \neq 0 \; . \qquad \text{(C-17)}$$

Now we have shown that $(1 + qs)G(s) + 1/K$ is positive real for all positive K; therefore we can choose K large enough for (C-17) to hold so long as $|Ih^o/26\, K_m| < 1$. Thus, the system is

globally asymptotically stable so long as $\beta_{1c} = 0$, $\gamma_{1c} - \gamma_{2c} \neq 0$, the gain is positive, and the initial total momentum is within the system capacity.

For the case $\beta_{1c} \neq 0$ the pitch and yaw channels are still globally asymptotically stable by the arguments above, but the roll channel cannot be handled as above because it has inputs from pitch and yaw. This situation is in the class covered by specializing the work of Sandberg [17].

Theorem (Sandberg). — Let $t^p w_{11}(t) \in \mathcal{L}_1(0,\infty) \cap \mathcal{L}_2(0,\infty)$ $p = 0,1,2$. Let $\rho(t) = e(t) + \int_0^\infty w_{11}(t-\tau) f_1^a(e(\tau))d\tau$, $t > 0$ where $\rho(t) \in \mathcal{L}_{\infty N}(0,\infty)$. $\big(\rho(t)$ is measurable and bounded on $(0,\infty)$ and $e(t) \in \mathcal{L}_N\big)\big(e$ is measurable on $(0,\infty)$ and all its time truncations are square integrable$\big)$. Let

$$w_{11}(s) = \int_0^\infty w_{11}(t)\, e^{-st}\, dt\,, \qquad \text{Re } s \geq 0\,.$$

Suppose that

i) $\left(1 + \frac{1}{2}(\alpha + \beta) w_{11}(s)\right) \neq 0$ for $\text{Re } s \geq 0$

ii) $\frac{1}{2}(\beta - \alpha) \sup_{-\infty < \omega < \infty} \left|\left(1 + \frac{1}{2}(\alpha+\beta) w_{11}(i\omega)\right)^{-1} w_{11}(i\omega)\right|^2 < 1$

Then $e(t) \in \mathcal{L}_{\infty N}(0,\infty)$, there exists a constant c, which depends only on w, α, β, such that

$$\sup_{t \geq 0} |e(t)| \leq c \sup_{t \geq 0} |\rho(t)|$$

and $e(t) \to 0$ as $t \to \infty$ whenever $\rho(t) \to 0$ as $t \to \infty$. Note that $\alpha \leq f_1^a(u_1)/u_1 \leq \beta$ and $f_1^a(0) = 0$ are also required.

The theorem is easily related to the diagram below where $\rho(t)$ is seen to be the input to the roll channel from the pitch and yaw

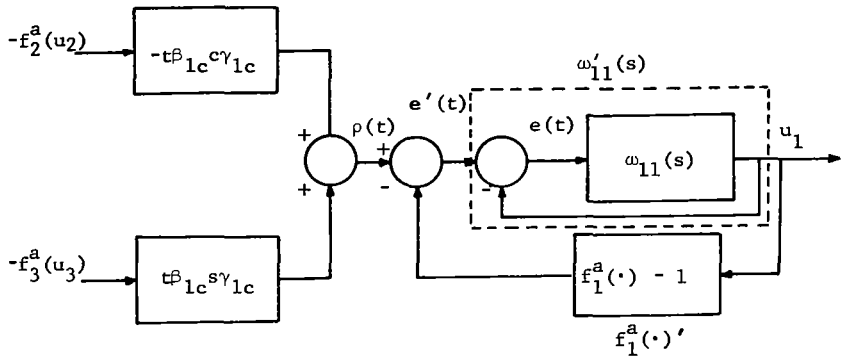


Fig. C-1 Illustration of System in Form for Application of a Sandberg Theorem

channels and $e(t)$ is the roll channel error signal. Unfortunately $w_{11}(s)$ contains a pole at the origin and therefore $tw_{11}(t)$ is not $\mathcal{L}_1(0,\infty)$ or $\mathcal{L}_2(0,\infty)$. That is why the system has been drawn as shown to create $w'_{11}(s) = w_{11}(s)\,(w_{11}(s) + 1)^{-1}$ which does satisfy this condition because all poles of $w'_{11}(s)$ are well into the left-half plane. It is easy to show by root locus arguments that condition i) holds for $\alpha = -1$, $\beta = 0$ which are required for $f_1^a(u_i)'$, the redefined nonlinearity. Condition ii) turns out to be

$$\tfrac{1}{2}(\beta-\alpha)\max_{\omega}\left|w'_{11}(i\omega)\left(1+\tfrac{1}{2}(\alpha+\beta)w'_{11}(i\omega)\right)^{-1}\right|^2 = \tfrac{1}{2}\max_{\omega}\left|\frac{w_{11}(i\omega)}{1+\tfrac{1}{2}w_{11}(i\omega)}\right|^2 < 1 \qquad \text{(C-18)}$$

but, unfortunately, it is easy to see that the expression has value 1 at $\omega = 0$. Therefore, the result (global stability for the coupled system) cannot be established this way.

The only remaining alternative is to try the very general result of Yacubovich [18]. (Brockett's theorem, given above, appears to be a special case of this theorem.) His work is apparently the most encompassing frequency domain stability result. From it one

can obtain the Popov theorem, the circle criterion, and results that are stronger than those of Popov. The matter is too complicated to present in detail, but in essence it is that the system must have a Hurwitz linear part (all eigenvalues strictly in the left-half plane); then one forms quadratic forms in the nonlinearities and their arguments that are nonnegative, and a complicated quadratic form is developed from the system function and the quadratic forms in the nonlinearities. If this quadratic form is negative definite in the nonlinearities (i.e., the $f_i^a(u_i)$ are considered as variables) then the system solution goes to zero with $t \rightarrow \infty$ and it is square integrable.

The matrix system was reformulated to fit the conditions of the theorem by extracting unit gain from each saturation and using it to shift the system poles as illustrated in Fig. C-1. We were not able to satisfy the other conditions of the theorem despite the variety of formulations of the Yacubovich function that were tried. Thus, we have to conclude that the Popov approximation is likely to be absolutely stable on the basis of single channel evidence and a visceral intuition; however, the most powerful frequency domain theorem could not be used to prove this. Further, the computations required by the Yacubovich theorem are very difficult and tedious to do by hand. This indicates that as it now stands some computational aid is required to make its application to large systems feasible and practical.

APPENDIX D

SOLUTIONS OF COMPUTATIONAL PROBLEMS

(i) Solution of the Liapunov Matrix Equation

A summary of four alternative methods for computing the solution to the Liapunov matrix equation will 1l be presented. Of the four algorithms to be discussed, the first has been described in Geiss et al. [19]; the second, third and fourth are based on the work of Ma [20], Smith [21], and Jameson [22], respectively.

Algorithm 1. — This is a reliable brute force approach in which the Liapunov equation, viz.,

$$A^T P + PA = - Q \tag{D-1}$$

is to be solved for P, A and Q being given. Here P and Q are assumed to be $n \times n$ and assumed to be stable throughout (the stability criterion is not necessary for algorithm 1).

Expanding (D-1) and rewriting it as a vector equation, we obtain

$$A_{mod} \begin{bmatrix} p_{11} \\ \vdots \\ p_{1n} \\ p_{21} \\ \vdots \\ p_{2n} \\ \vdots \\ p_{nn} \end{bmatrix} = - \begin{bmatrix} q_{11} \\ \vdots \\ q_{1n} \\ q_{21} \\ \vdots \\ q_{2n} \\ \vdots \\ q_{nn} \end{bmatrix}, \tag{D-2}$$

where A_{mod} is $n^2 \times n^2$ and is given by

$$A_{mod} = A^T \circledast I_n + I_n \circledast A^T \tag{D-3}$$

in which I_n is the $n \times n$ identity matrix, and $\circledast$ is the Kronecker product [23]. For $n = 2$ we have

$$A_{mod} = \left[\begin{array}{c|c} A^T + a_{11}I_2 & a_{21}I_2 \\ \hline a_{12}I_2 & A^T + a_{22}I_2 \end{array}\right] . \tag{D-4}$$

The solution of (D-2) is then simply a matter of inverting A_{mod} and reconstructing the P matrix. One can readily see the computational difficulties that would arise in attempting to solve the original Liapunov equation for dimensions much above 10. Inversion of matrices well above 100×100 would be necessary. If only the core storage of a large machine is used, matrix inversion is limited to matrices less than 150×150.

Three rather more elegant developments follow.

Algorithm 2. — The matrix equation, (D-1), can be solved for P as follows: Assume that

$$(\lambda - \lambda_i)^{\alpha_i} \quad , \quad i = 1, 2, \ldots, r \quad , \quad \sum_{i=1}^{r} \alpha_i = n \tag{D-5}$$

are the elementary divisors of A (and thus A^T) over $\mathbb{C}$, the field of complex numbers. Then there exist matrices U, V such that

$$A = U^{-1}\tilde{A}_1 U \tag{D-6}$$

and

$$A^T = V^{-1}\tilde{A}_1 V \ , \tag{D-7}$$

where $\tilde{A}_1$ is the Jordan normal form for A (and thus A^T); i.e.,

$$\tilde{A}_1 = \text{diag}\left\{\lambda_1 I_{\alpha_1} + N_{\alpha_1},\ \lambda_2 I_{\alpha_2} + N_{\alpha_2},\ \ldots,\ \lambda_r I_{\alpha_r} + N_{\alpha_r}\right\} \quad \text{(D-8)}$$

$$I_{\alpha_\mu} = (\delta_{ij}) \quad , \quad i,j = 1, 2, \ldots, \alpha_\mu \quad \text{(D-9)}$$

$$N_{\alpha_\mu} = (\delta_{i+1,j}), \quad i,j = 1, 2, \ldots, \alpha_\mu \quad \text{(D-10)}$$

and δ_{ij} is the Kronecker delta. The Liapunov equation then becomes

$$\tilde{A}_1 Y + Y\tilde{A}_1 = - D \ , \quad \text{(D-11)}$$

where

$$Y = VPU^{-1} \quad \text{(D-12)}$$

and

$$D = VQU^{-1} \ . \quad \text{(D-13)}$$

In [20], Ma gives a finite series solution for the matrix equation

$$\tilde{A}X - X\tilde{B} = C \ , \quad \text{(D-14)}$$

where $\tilde{A}$ and $\tilde{B}$ are in Jordan normal form. Thus, via the identification $X = Y$, $\tilde{A} = \tilde{A}_1$, $\tilde{B} = - \tilde{A}_1$, and $C = - D$, the solution to (D-11) is obtained from Ma's solution to (D-14). The solution is:

$$Y_{ij} = - \sum_{n=0}^{\alpha_i+\alpha_j-2} (\lambda_i + \lambda_j)^{-(n+1)} (-1)^n \sum_{\sigma+\tau=n} \frac{n!}{(n-\tau)!\,\tau!} N_{\alpha_i}^{\sigma} D_{ij} N_{\alpha_j}^{\tau} \ , \quad \text{(D-15)}$$

where Y_{ij} and D_{ij} are the ij elements of the partitions of Y and D, which are the same as the partition of $\tilde{A}_1$. Finally, assuming that $\tilde{A}_1$, V, and U can be computed, we obtain P as

$$P = V^{-1}YU \tag{D-16}$$

from (D-12). Note that since A is assumed to be stable, all λ_i will have negative real parts and

$$\lambda_i + \lambda_j \neq 0 \qquad i,j = 1, 2, \ldots, r \ . \tag{D-17}$$

Thus only the first case of Ma's solution, (D-15), need be considered here. Further, if A is of simple structure (i.e., the eigenvalues are distinct), then the solution is

$$y_{ij} = \frac{-d_{ij}}{\lambda_i + \lambda_j} \ , \qquad i,j = 1, 2, \ldots, n \ , \tag{D-18}$$

where y_{ij} and d_{ij} are elements of Y and D, respectively.

The main drawback of this method is the requirement that A be provided in Jordan normal form, along with the appropriate transformation matrices U and V. There does not appear to be any reliable computational procedure for doing this for arbitrary matrices, in particular for those of the degenerate eigenvector, multiple eigenvalue variety.

Algorithm 3. — Smith [21] presents a scheme for obtaining an explicit expression for P in (D-1) that uses the n^2 elements of A in the relatively simple operations of matrix multiplication and addition. In the more difficult operations of determinantal expansion, only n variables are involved.

Define constants $k_1, \ldots, k_n$, and an $n \times n$ matrix G by:

$$\det(\lambda I - A) = \lambda^n + k_1\lambda^{n-1} + k_2\lambda^{n-2} + \ldots + k_n \tag{D-19}$$

$$G = \begin{bmatrix} 0 & 1 & 0 & 0 & \ldots & 0 \\ 0 & 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 0 & 1 & \ldots & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & \ldots & 1 \\ -k_n & -k_{n-1} & -k_{n-2} & -k_{n-3} & \ldots & -k_1 \end{bmatrix} \tag{D-20}$$

THEOREM 1 (Smith). — If Z is any $n \times n$ matrix, and $U = (u_{ij})$, $C = (c_{ij})$ are $n \times n$ matrices satisfying

$$G^T U + UG = - C , \tag{D-21}$$

then the following matrices satisfy (D-1):

$$P = \sum_{i=1}^{n} \sum_{j=1}^{n} u_{ij} (A^T)^{i-1} Z A^{j-1} \tag{D-22}$$

$$Q = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} (A^T)^{i-1} Z A^{j-1} . \tag{D-23}$$

COROLLARY (Smith). — If $U = (\hat{u}_{ij})$ is a solution of (D-21) when $C = \mathrm{diag}(c, 0, \ldots, 0)$, then

$$P = \sum_{i=1}^{n} \sum_{j=1}^{n} \hat{u}_{ij} (A^T)^{i-1} Z A^{j-1} \tag{D-24}$$

is a solution of

$$A^T P + PA = - cZ . \tag{D-25}$$

If $c \neq 0$, setting $Z = c^{-1} Q$ in (D-24) and (D-25), we obtain

$$P = \frac{1}{c} \sum_{i=1}^{n} \sum_{j=1}^{n} \hat{u}_{ij} (A^T)^{i-1} Q A^{j-1} , \tag{D-26}$$

which is the solution of

$$A^T P + PA = - Q . \tag{D-27}$$

It remains to be shown how one obtains the $(\hat{u}_{ij})$ elements to be used in (D-26). Let us define the following matrix:

$$H \overset{\Delta}{=} \begin{bmatrix} k_1 & k_3 & k_5 & k_7 & \cdots & k_{2n-1} \\ k_o & k_2 & k_4 & k_6 & \cdots & k_{2n-2} \\ 0 & k_1 & k_3 & k_5 & \cdots & k_{2n-4} \\ 0 & k_o & k_2 & k_4 & \cdots & k_{2n-4} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & .. & .. & \cdots & k_n \end{bmatrix} \qquad \text{(D-28)}$$

where $k_o = 1$ and $k_p = 0$ for all $p > n$.

THEOREM 2 (Smith). — If $k_1, \ldots, k_n$ are real, then the equation

$$G^T V + VG = - \operatorname{diag}(h, 0, \ldots, 0) \qquad \text{(D-29)}$$

has the hermitian solution $V = (v_{\alpha\beta})$ given by

$$v_{\alpha\beta} = \frac{1}{2}(-1)^{\alpha-1} \sum_{r=0}^{n-\alpha} \sum_{s=0}^{n-\beta} (-1)^r k_r k_s \phi\left(\frac{1}{2}[\alpha+\beta+r+s]\right) , \qquad \text{(D-30)}$$

where $h = \det(H)$, $k_o = 1$, $\phi(\xi) = 0$, when ξ is not an integer; otherwise $\phi(\xi)$ is the cofactor of the element in the first row and ξ^{th} column of the H matrix.

We see that if $c = h$, then the solution for V given by (D-30) can be substituted into (D-26), where $u(\alpha,\beta) = v(\alpha,\beta)$, $\alpha = 1, \ldots, n$ $\beta = 1, \ldots, n$.

We shall now show a convenient way of determining the numbers $k_1, \ldots, k_n$ without determinantal expansion. First, we define

$$s_v = \operatorname{trace}[A^v] .$$

Since $k_1, \ldots, k_n$ are homogeneous polynomials in the eigenvalues of A, they can be written via Newton's identities in terms of $s_1, s_2, \ldots, s_n$.

$$
\begin{aligned}
& s_1 + k_1 = 0 \\
& s_2 + s_1 k_1 + 2k_2 = 0 \\
& s_3 + s_2 k_1 + s_1 k_2 + 3k_3 = 0 \\
& \cdots\cdots\cdots\cdots \\
& s_n + \cdots\cdots nk_n = 0 \ .
\end{aligned}
\tag{D-31}
$$

This method of obtaining the k's is attractive because the computation of A^v for $v = 1, 2, \ldots, n$ is already required for the computation of P in (D-26).

The cofactors of H are relatively easy to calculate since the matrix involves only the variables $k_1, \ldots, k_n$, and many of its elements are zero

Algorithm 4. — Jameson [22] has independently devised a technique for numerically computing the solution to

$$AX + XA^T = C \tag{D-32}$$

where A, C, and X are real $n \times n$ matrices. C is generally symmetric, in which case X is also symmetric.

Form the sequence of expressions:

$$
\begin{aligned}
& C_o = 0 \\
& C_1 = C = AX + XA^T \\
& C_2 = AC_1 - C_1A^T + AC_oA^T = A^2X - XA^{T^2} \\
& \cdots\cdots\cdots\cdots\cdots\cdots \\
& C_N = AC_{N-1} - C_{N-1}A^T + AC_{N-2}A^T = A^NX - (-1)^N XA^{T^N} \ .
\end{aligned}
\tag{D-33}
$$

The recurrence relation may also be written as

$$C_k = A^{k-1}C - C_{k-1}A^T \tag{D-34}$$

or

$$C_k = AC_{k-1} - CA^{T^{k-1}} . \tag{D-35}$$

If C is symmetric, then C_k is symmetric when k is odd and antisymmetric when k is even. Let the characteristic equation of A be

$$|\lambda I - A| = \lambda^N + a_1\lambda^{N-1} \ldots + a_N = (\lambda - \lambda_1)(\lambda - \lambda_2) \ldots (\lambda - \lambda_N) = 0 \tag{D-36}$$

According to the Cayley Hamilton theorem this is satisfied by A and A^T. Therefore

$$C_N - a_1C_{N-1} \ldots + (-1)^{N-1}a_{N-1}C_1 = A^NX - a_1A^{N-1}X \ldots + (-1)^{N-1}a_{N-1}AX + (-1)^Na_NX \tag{D-37}$$

$$C_N + a_1C_{N-1} \ldots + a_{N-1}C_1 = -a_NX - (-1)^N\left[XA^{T^N} - a_1XA^{T^{N-1}} \ldots + (-1)^{N-1}a_{N-1}XA^T\right] .$$

Thus the solution of (D-32) may be written as

$$X = G^{-1}K = L(G^T)^{-1} , \tag{D-38}$$

where

$$G = A_N - a_1A^{N-1} \ldots + (-1)^Na_NI = (A + \lambda_1 I)(A + \lambda_2 I) \ldots (A + \lambda_N I) \tag{D-39}$$

$$K = C_N - a_1C_{N-1} \ldots + (-1)^{N-1}a_{N-1}C_1 \tag{D-40}$$

$$L = C_N + a_1C_{N-1} \ldots + a_{N-1}C_1 . \tag{D-41}$$

If C is symmetric then $L = K^T$. Since the determinant of a product of matrices is equal to the product of their determinants, it is evident that G is not invertible if for any i, $-\lambda_i$ is a characteristic value of A. The equation can therefore be solved if and only if $\lambda_i + \lambda_j \neq 0$ for all i,j.

The characteristic coefficients a_i may be determined by Bocher's identities [25].

$$
\begin{aligned}
a_1 &= \mathrm{tr}(A) \\
a_2 &= -\frac{1}{2}\left[a_1 \mathrm{tr}(A) + \mathrm{tr}(A^2)\right] \\
&\cdots\cdots\cdots\cdots\cdots\cdots \\
a_N &= -\frac{1}{N}\left[a_{N-1}\mathrm{tr}(A) + a_{N-2}\mathrm{tr}(A^2) \ \ldots + \mathrm{tr}(A^N)\right] = (-1)^N |A| \ .
\end{aligned}
\tag{D-42}
$$

Alternatively, they may be determined from the characteristic coefficients by the rule for polynomial multiplication.

By adding and subtracting the characteristic equation for A to (D-39), G can be expressed in terms of even or odd powers only of A,

$$G = 2\left[r_{11}A^N + r_{12}A^{N-2} \ \ldots\right] \tag{D-43}$$

$$= -2\left[r_{21}A^{N-1} + r_{22}A^{N-3} \ \ldots\right] , \tag{D-44}$$

where

$$
\begin{aligned}
&r_{11} = 1, \quad r_{12} = a_2, \ r_{13} = a_4, \ \ldots \\
&r_{21} = a_1, \ r_{22} = a_3, \ r_{23} = a_5, \ \ldots \ .
\end{aligned}
\tag{D-45}
$$

If (D-44) is multiplied by $(r_{11}/r_{21})A$ and subtracted from (D-43), then A^N is eliminated, yielding

$$\left[I + \frac{r_{11}}{r_{21}} A\right] G = 2\left[r_{31}A^{N-2} + r_{32}A^{N-4} \ \ldots\right] , \tag{D-46}$$

where the coefficients r_{ij} are generated by Routh's rule

$$r_{ij} = r_{i-2,j+1} - r_{i-1,j+1} \frac{r_{i-2,1}}{r_{i-1,1}} . \tag{D-47}$$

A^{N-1} can then be eliminated between (D-44) and (D-46), and this procedure can be repeated until the right hand side is finally reduced to the identity matrix. It follows that

$$G^{-1} = \frac{(-1)^N}{2r_{N+1,1}} H_{N+1} , \tag{D-48}$$

where

$$\begin{aligned} H_1 &= I \\ H_2 &= I \\ H_3 &= H_1 + \frac{r_{11}}{r_{21}} AH_2 \\ &\cdots\cdots\cdots\cdots \\ H_{N+1} &= H_{N-1} + \frac{r_{N-1,1}}{r_{N,1}} AH_N . \end{aligned} \tag{D-49}$$

The inverse of G is thus expressed as a power series in A up to the $(N-1)^{th}$ power.

The Liapunov equation, (D-1), may thus be solved by using the result, (D-38), and

a) forming G^{-1} directly according to (D-48)

or

b) forming G according to (D-39) and inverting the resulting system of equations.

Method b), which has been found to be computationally superior, is the one incorporated in the program.

The four algorithms discussed in this report have all been programmed for use on the IBM 360-75, and have given satisfactory results in solving the Liapunov equation.

Input data for both an eight and nine dimensional problem were used in the trial runs, with the elements of the A matrix for $n = 8$ being taken from the dynamical equations of the U.S. Naval aircraft E2A lateral control system. The A matrix for $n = 9$ was chosen out of convenience, as one for which the Jordan form and transformation matrices were known.

PROGRAM 1 (Full Inversion): Requires problem dimension (n), Q, and A as inputs. Computing time for a nine dimensional problem = 1.22 seconds, with t proportional to $(n)^6$.

PROGRAM 2 (Er-Chieh Ma Method): Requires n, Q, A, the elements of the elementary divisors of A (and thus A^T), and the transformation matrices U, U^{-1}, V, V^{-1} as inputs. Computing time for a nine dimensional problem = 0.11 seconds, with t proportional to $(n)^5$. Note that the computation of the transformation matrices for a general nonsymmetric matrix, A, is nontrivial.

PROGRAM 3 (Smith's Method): Requires n, Q, and A as inputs. Computing time for a nine dimensional problem = 1.06 seconds, with t proportional to $(n)^5$.

PROGRAM 4 (Jameson's Method): Requires n, Q, and A as inputs. Computing time for a nine dimensional problem = 0.5 seconds, with t proportional to $(n)^4$.

So far as accuracy is concerned, all the methods give comparable results for the dimensions we have used (up to $n = 9$). Accuracy deterioration increases with dimension for all programs, but the one that suffers least in this respect is Program 1 (full inversion method), which is also the one that suffers most with respect to increase of computing time with dimension.

It appears that, faced with the problem of employing a computer program for solving the Liapunov equation, one should choose between Program 1 (full inversion) and Program 4 (Jameson's method).

The full inversion method suffers least in accuracy deterioration with dimension increase, while the latter is superior in terms of computing time variation with dimension. Program 2 (Ma's method) is not particularly practical in that it requires the Jordan form of A as well as the transformation matrices as inputs. There does not exist at present a program that will find the eigenvalues and eigenvectors for any general nonsymmetric matrix with a reasonable degree of reliability. Even if one did exist and was used as a subroutine for Program 2, the total computing time required would far exceed the 0.11 seconds for $n = 9$ reported here.

It should be noted that Jameson's method, although it was developed independently, is essentially a refinement of Smith's approach. Both methods are more prone to numerical inaccuracy for the case of ill-conditioned A matrices than the other two approaches. Program 4 (Jameson's method), which utilizes triple precision accumulation, seems to have alleviated this problem and thus is superior to Program 3 (Smith's method) for which only double precision was feasible.

(ii) Parameterization of the Set of Positive Definite Matrices and an Algorithm for Generation of Its Elements

This section describes an efficient algorithm that generates arbitrary $n \times n$ positive definite symmetric matrices, as required in Section 3. These matrices were used as candidate "Q" matrices, which in turn generated "P" matrices through the solution of the Liapunov equation, $A^TP + PA = -Q$. Having parameterized the Q set, one can proceed in orderly fashion in pursuit of an "optimal" quadratic form Liapunov function to resolve the domain of stability problem.

The generation of the set of positive-definite $n \times n$ symmetric matrices can be carried out by resorting to the brute force approach of forming an arbitrary $n \times n$ symmetric matrix and then applying the determinantal test [26] for positive-definiteness. The arbitrary choice of $n(n + 1)/2$ matrix elements followed by the evaluation of the determinants of the n-principal minors would be necessary. It would be desirable to generate these matrices by a procedure that guarantees all to be positive-definite, and in addition, that the entire set of positive-definite matrices be spanned.

It is well known [26] that all real symmetric matrices are orthogonally similar to a diagonal matrix, and that all positive-definite (pd) matrices are then orthogonally similar to a diagonal matrix with positive-diagonal elements; i.e., let Q be pd, then

$$Q = S^T \Lambda S \ , \tag{D-50}$$

where

$$\Lambda = \text{diag} \left\{ \lambda_1, \ \lambda_2, \ \ldots, \ \lambda_n \right\}$$

$$\lambda_i > 0 \ , \qquad i = 1, \ 2, \ \ldots, \ n \tag{D-51}$$

$$S^T S = I \ .$$

Thus, the parameterization of all pd matrices, Q, is reduced to the parameterization of the group of orthogonal matrices, S.

In [9], Murnaghan proves that the parameterization of the group of $n \times n$ unitary matrices U is accomplished by the factorization

$$U = D \left[\prod_{k=1}^{n-1} U_{n-k} \right] = D \times U_{n-1} \times \ldots \times U_1 \ , \tag{D-52}$$

where

$$D = \text{diag} \left\{ e^{i\delta_1}, \ e^{i\delta_2}, \ \ldots, \ e^{i\delta_{n-1}}, \ e^{i\phi_n} \right\} \tag{D-53}$$

$$U_k = \left[\prod_{\ell=k+1}^{n-1} U_{k\ell}(\theta_\mu, \ \sigma_\rho) \right] \left[U_{kn}(\phi_k, \ \sigma_\gamma) \right] \ ,$$

$$\gamma = \frac{(2n - k)(k - 1)}{2} + 1 \ ,$$

$$\rho = \frac{(2n - k)(k - 1)}{2} + 1 + n - \ell \ , \tag{D-54}$$

$$\mu = \frac{(2n - k - 2)(k - 1)}{2} + (n - \ell) \ ,$$

$$u_{ii} = 1, \quad i \neq k, \ell$$

$$u_{kk} = \cos\theta$$

$$u_{\ell\ell} = \cos\theta$$

$U_{k\ell}(\theta, \sigma) = (u_{ij})$: (D-55)

$$u_{ij} = 0, \quad i \neq j, \quad i,j \neq k, \ell$$

$$u_{k\ell} = -e^{-i\sigma} \sin\theta$$

$$u_{\ell k} = +e^{+i\sigma} \sin\theta \ ,$$

$$-\pi \leq \phi < \pi \ , \quad -\frac{\pi}{2} \leq \sigma \leq \frac{\pi}{2} \ , \quad -\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2} \ , \quad -\frac{\pi}{2} \leq \delta \leq \frac{\pi}{2} \ .$$

The factorization of the group of orthogonal matrices is immediately obtained by requiring U to be real; i.e., $\delta = \sigma = 0$, $\phi_n = \pm\pi$, $-\pi \leq \phi_k < \pi$, $k \neq n$, and $-\pi/2 \leq \theta \leq \pi/2$. In particular,

$$S = D_1 \left[\prod_{k=1}^{n-1} S_{n-k} \right] , \tag{D-56}$$

$$D_1 = \text{diag} \left\{ 1, \ \ldots, \ 1, \ \pm 1 \right\} , \tag{D-57}$$

$$S_k = \left[\prod_{\ell=k+1}^{n-1} S_{k\ell}(\theta_\mu) \right] \left[S_{kn}(\phi_k) \right] \quad , \quad S_{k\ell}(\theta_\mu) = U_{k\ell}(\theta_\mu, 0) \tag{D-58}$$

$$\mu = \frac{(2n - k - 2)(k - 1)}{2} + n - \ell \ .$$

This factorization contains $(n - 1)(n - 2)/2$ thetas and n phis, or a total of $[n(n - 1)/2] + 1$ parameters. The n lambdas in (D-51) raise the number of parameters to $[n(n + 1)/2] + 1$, or one more than required. Thus, if we restrict S to be a rotation

matrix (i.e., choose $\phi_n = 0$), the number of parameters will be $n(n+1)/2$, the number required to represent an arbitrary symmetric matrix. The choice $\phi_n = 0$ is made into order to rotate and scale the ellipsoid associated with the quadratic form formed from the pd matrix and without reflecting coordinates or permuting the coordinate system.

The factorization of a pd matrix of dimension three is thus given by

$$P = S^T \Lambda S \tag{D-59}$$

where

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \tag{D-60}$$

$$\lambda_1, \lambda_2, \lambda_3 > 0$$

and

$$S = S_2 S_1 = S_{23}(\phi_2) S_{12}(\theta_1) S_{13}(\phi_1) , \tag{D-61}$$

$$S_{23} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi_2 & -s\phi_2 \\ 0 & s\phi_2 & c\phi_2 \end{bmatrix}$$

$$S_{12} = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 \\ s\theta_1 & c\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{D-62}$$

$$S_{13} = \begin{bmatrix} c\phi_1 & 0 & -s\phi_1 \\ 0 & 1 & 0 \\ s\phi_1 & 0 & c\phi_1 \end{bmatrix} ,$$

$$-\pi \leq \phi_1 < \pi, \quad -\pi \leq \phi_2 < \pi, \quad -\frac{\pi}{2} \leq \theta_1 \leq \frac{\pi}{2} ,$$

$$c\phi_1 = \cos \phi_1, \quad s\phi_1 = \sin \phi_1 .$$

Thus, it is clear that by using this representation under the restrictions

$$\begin{aligned} &\lambda_i > 0 \quad , && i = 1, 2, \ldots, n \\ &-\pi \leq \phi_i < \pi \quad , && i = 1, 2, \ldots, n-1 \\ &-\frac{\pi}{2} \leq \theta_i \leq \frac{\pi}{2} \quad , && i = 1, 2, \ldots, \frac{(n-1)(n-2)}{2} \end{aligned} \tag{D-63}$$

the candidate Q matrices are guaranteed to be positive definite. For ease in programming the constraints, (D-63) were removed by defining

$$\begin{aligned} &\lambda_i = e^{\eta_i} \quad , && \eta_i \text{ real} \\ &\theta_i = \tfrac{1}{2}(-\pi + \theta_i' \bmod 2\pi) \quad , && \theta_i' \text{ real} \end{aligned} \tag{D-64}$$

and recognizing that since the ϕ_i's only appear as arguments of trigometric functions, they can be arbitrary real numbers.

(iii) Penalty Function Formulation and Gradient Search

The problem is to determine ℓ where

$$\begin{aligned} &\ell = \min_{x \in E} V(x) \\ &E: \left(x \mid \dot{V}(x) = 0 \ , \ x \neq 0\right) \end{aligned} \tag{D-65}$$

using gradient search. The penalty function formulation is used to convert the constrained minimum problem, wherein the constraint surface $\left(\dot{V}(x) = 0\right)$ is determined by the nonlinear part $\left(g(x)\right)$ of the original problem $\left(\dot{x} = Ax + g(x)\right)$ to an unconstrained problem. This approach changes the constrained problem to the unconstrained problem where ℓ is now redefined as

$$\ell = \min_{x}\left[\left(V(x)\right)^{n_1} + k_1\left(\dot{V}(x)\right)^2 + k_2\rho(x)\right] \qquad \text{(D-66)}$$

where $k_1 > 0$, $k_2 > 0$, $n_1 = 1$ or 2, $\left(\dot{V}(x)\right)^2$ is the penalty for straying from $\dot{V}(x) = 0$, and $\rho(x)$ is the penalty for approaching $x = 0$.

All gradient search procedures require the computation of the gradient of the function to be minimized either analytically or numerically. The analytical gradient of this problem was extremely complex to calculate and the calculations were prone to human error. The numerical gradient computation was beset with step size problems. Step size was critical in that a single fixed step size was not applicable in all coordinate directions; incorrect zero gradients were obtained in some directions. An adaptive step size was tried, but the function was still too varied in each direction, and finally the analytical gradient was resorted to after assurance of its accuracy by three different persons' doing independent calculations. The numerical gradient never agreed exactly (to six significant figures) with the analytical gradient and it was decided to accept the analytical gradient.

The minimization algorithm utilized was "MIN-ALL" [27], a conjugate gradient procedure developed at Grumman Aerospace Corporation based on the work of Davidon [28] and Fletcher and Powell [29] to solve for the unconstrained ℓ utilizing various combinations of k_1, k_2, and n_1.

This procedure required 20 minutes of IBM 360/75 computer time to evaluate ℓ given Q, since the evaluation of a gradient was required at every point along the search path. The lengthy computation was a drawback, but the frequency of arriving at a local minimum (even with the penalty functions) was far too high. Since this procedural search was also intended for the 45-parameter space search for the optimal Q-matrix, and this much uncertainty and difficulty, not to mention the computational time, had been encountered, it was a definite requirement to find another organized search procedure.

(iv) Random Search Solution of the Minimum Problem

Given the matrix pair Q, P satisfying the Liapunov equation

$$-Q = A^T P + PA \qquad \text{(D-67)}$$

where Q, P are positive definite symmetric matrices, and A is stable. Define

$$V(x) = x^T P x \tag{D-68}$$

so that

$$\dot{V}(x) = - x^T Q x + 2x^T P g(x) \tag{D-69}$$

where $g(x)$ is $o(\|x\|)$. The problem considered in this appendix section is the determination of

$$\ell = \min_{x \in E} V(x) \tag{D-70}$$

where

$$E = \left\{ x \,|\, \dot{V}(x) = 0 \ , \quad x \neq 0 \right\} . \tag{D-71}$$

A random search algorithm was developed as a more effective procedure to cover the state space more uniformly and avoid local minimum problems. It utilizes the fact that if for a particular point x_1 in the n dimensional state-space with corresponding Liapunov function value $c_1 = V(x_1) = x_1 P x_1^T$ we have $\dot{V}(x_1) > 0$; then the global minimum of $V(x)$ on $\dot{V}(x) = 0$ must lie inside the ellipsoid consisting of the set of states x satisfying $x^T P x \leq c_1$. The program thus randomly searches state points within this ellipsoid until it again finds a point x_2 corresponding to $V(x_2) = x_2 P x_2 = c_2$, $c_2 < c_1$, with $\dot{V} > 0$. This procedure is iterated until convergence is achieved. The longer the program runs, the higher the confidence in the closeness of the final value to the true global minimum.

The search begins by arbitrarily choosing a point (nine-tuple) in the state-space (see Fig. D-1 for a two dimensional version of this technique), and a large value of V, called V_o. ($V_o = 1$ is very large; V_o corresponds to r in the gradient search discussed previously.) The Liapunov matrix equation is still solved for P;

$$V = x^T P x \tag{D-72}$$

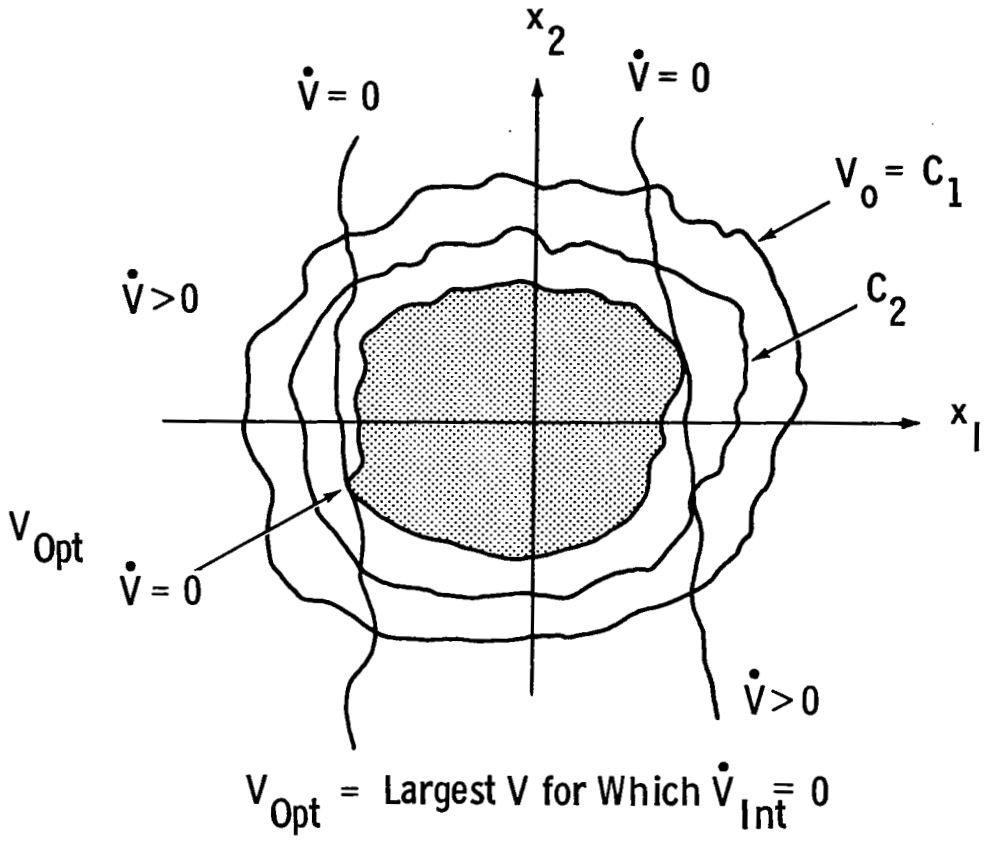


Fig. D-1 Two Dimensional Representation of Search Region

and

$$\dot{V} = - x^T Q x + 2 x^T P g(x) \quad . \tag{D-73}$$

It can be seen from the expression for V that the eigenvalues of P will determine the magnitude of the intercepts along the eigenvector directions by.

$$y_{i_{INT}} = \sqrt{V/\lambda_i} \qquad i = 1, 2, \ldots, 9 \ ,$$

where λ_i are the eigenvalues of P and V is a value of the function $V(x)$ at a particular point. A point in the x-space is related to a point in the y-space (eigenvector space) by a pure rotation given by

$$x = Cy \ , \qquad \text{(D-74)}$$

where C is the normalized matrix of eigenvectors of P (see Fig. D-2).

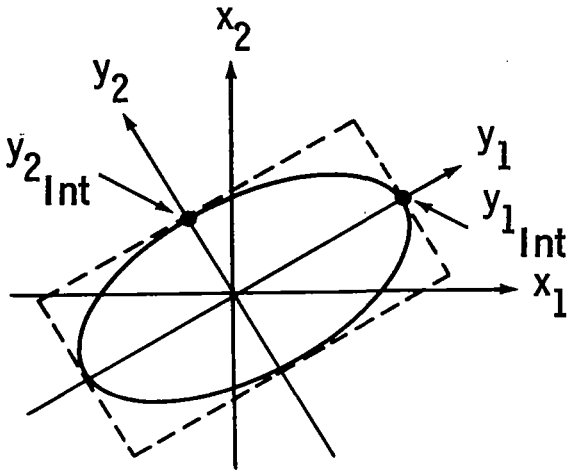


Fig. D-2 Relationship of State Space (x) to Its Associated Eigenvector Space (y) in Two Dimensions

Multiplying n random numbers constrained to the interval $[-1, +1]$ by the n intercept values $(y_{i_{INT}})$ yields a random vector interior to the hypercube containing $V(x) < V_{min}$. Transformation to the x-space allows a computation of V and $\dot{V}$. If $\dot{V} < 0$, the vector is discarded and a new random selection is made. If $\dot{V}(x) > 0$, and $V(x) < V_{min}$, a new ellipsoid with corresponding principal axes intercepts is computed, and the procedure is repeated using this new $V_{min} = V(x)$. The entire process is then iterated using the updated value of V_{min} until a good estimate of the global minimum is achieved with a high level of confidence. This is determined by the total number of random points selected in state space, as discussed below.

Experimental results indicated that when the random numbers were generated from a uniform distribution, a high percentage of the resulting V's were larger than the V calculated at the y-intercept point. In an attempt to compensate for this skewing effect, the eigenvector coordinates were generated such that the distribution near the boundaries would be attenuated. The Gaussian

distribution model was tried and proved quite successful. Each scaled vector component is generated independently via the same Gaussian model (all zero mean, same variance). This gives a Rayleigh type distribution in the scaled radial envelope, i.e.,

$$p(r) \approx \frac{r^{n-1} \exp(-r^2/2\sigma^2)}{k(\sigma, n)},$$

where σ^2 is the variance for the Gaussian distribution and n is the dimension of the space. The tail of the radial distribution can be attenuated as desired by varying σ. A further modification was made by introducing a switching function which changes σ after a certain number of iterations: for less than 1000 iterations, $\sigma = 1/6$, while between 1000 and 5000 iterative points $\sigma = 1/3$.

Further improvements in efficiency were achieved by coupling a linear bisection type search with the random search, as follows. If a point $x^{(1)}$ is found corresponding to $\dot{V}(x^{(1)}) > 0$, a one dimensional search along the line from $x^{(1)}$ to the origin is employed to find a point $kx^{(1)}$, $0 < k < 1$ such that $\dot{V}(kx^{(1)}) = 0$. Then the random search is continued in the smaller region $V(x) \leq V(k_1 x_1)$. This deterministic search is illustrated in Fig. D-3, and specifically for our program, proceeds to sequentially bisect intervals along the line $(0, x^{(1)})$ 15 times, which gives excellent convergence on a zero crossing of $\dot{V}(x)$ along the $x^{(1)}$ direction.

The number of points in any search was experimentally determined. Experience showed that even if up to 300,000 nine-tuples were selected, the best estimate usually occurred before 5000 points were selected. Thus 5000 points became the standard number of points x to be run during any one random search of state space.

To summarize briefly, all improvements in computing efficiency have refined the program to the stage where it is capable of getting good estimates of the global minimum of $V(x)$ on $\dot{V}(x) = 0$, $x \neq 0$ for a given Q in about 30 seconds computing time, with results comparable to earlier random search runs exceeding 25 minutes. Figure D-4 shows a flow diagram of the Random Search Technique.

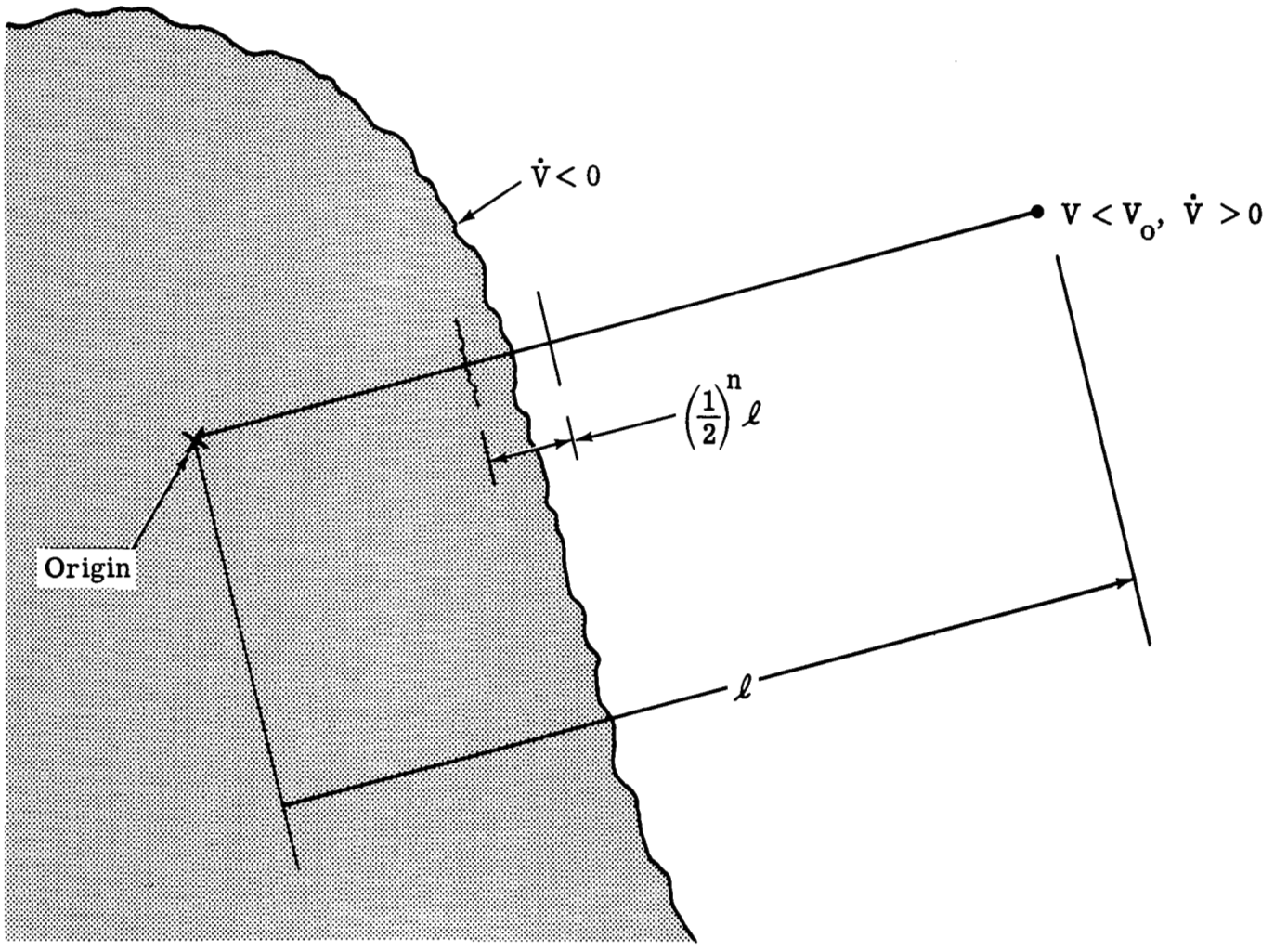


Fig. D-3 Schematic Representation of Biséction Deterministic Search

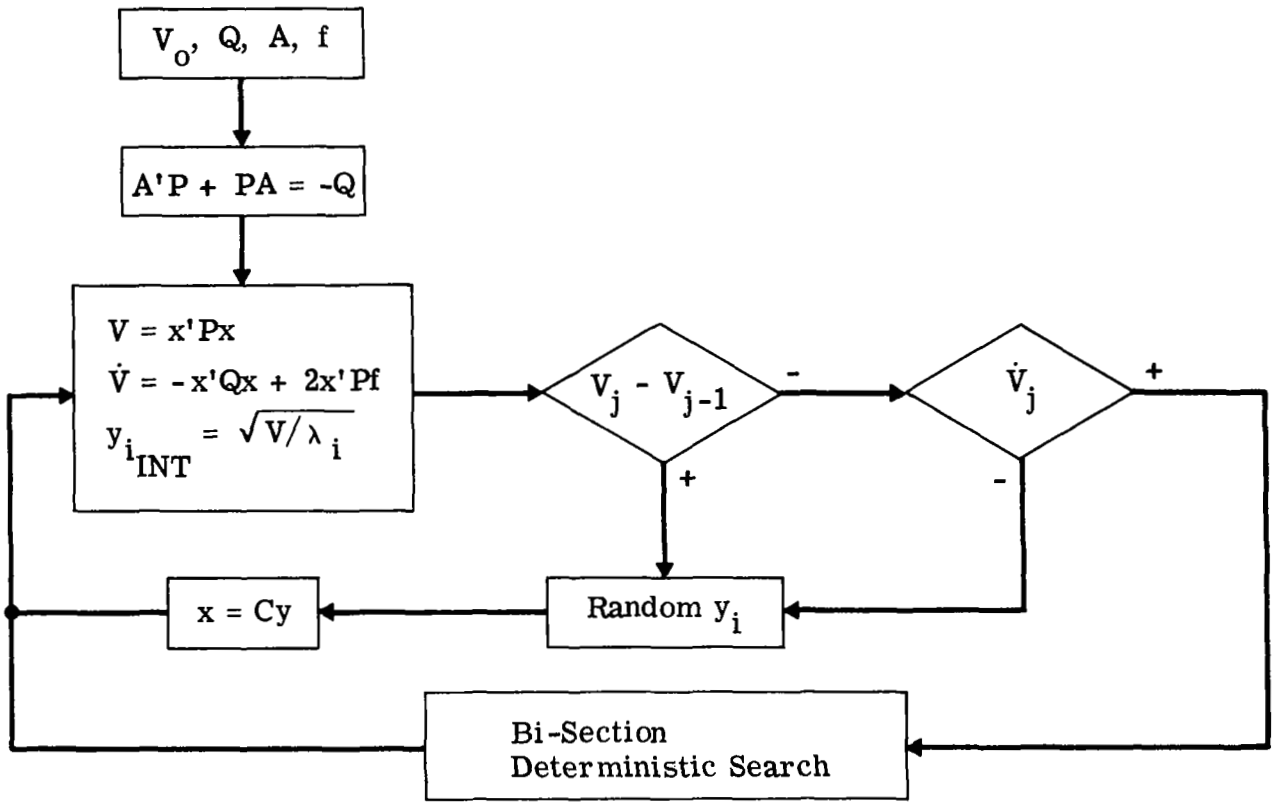


Fig. D-4 Schematic Flow Chart of the Random Search of the State Space

(v) A Random Search for the Maximum Problem

The random search of the 45-dimensional space of parameters from which the Q-matrix is determined is termed the maximum problem, since what is ultimately being sought is the maximum volume estimate of the domain of attraction by choice of Q.

An orderly method to search this space was required. Barron [10] gave some insight into this problem wherein he termed his method "an accelerated random search." Applied to this problem, the technique requires a starting point (45-tuple), in the space which is to be searched, from which is obtained an inverse volume estimate (a carry-over from the original formulation for application of a gradient minimization routine).

A performance measure $\underline{P}$ is defined as

$$\underline{P} = \log \left[\frac{(\text{vol}^{-1})}{10^{30}} \right] \tag{D-75}$$

where 10^{30} is arbitrary and chosen to keep $\underline{P}$ positive and in the vicinity of unity. Since this first estimate is the best to date, set $\underline{P}^*$, the best estimate, equal to $\underline{P}$. A random step, consistent with the parameter space constraints on λ, θ, and ϕ (i.e., $\lambda > 0$, $|\theta| \leq \pi/2$, $|\phi| < \pi$), is chosen by first defining

$$\sigma = \log \underline{P}^*$$

picking a random number χ, $-1 \leq \chi \leq +1$, for each parameter in the set λ, ϕ, θ and finally defining each step size as

$$\Delta u = (\text{sgn } \chi) \frac{L}{d} e^{-\chi^2/\sigma^2} \tag{D-76}$$

where

$$L = \begin{cases} 1000 & \text{for } \lambda_i \quad i = 1, 9 \\ \pi/2 & \text{for } \theta_i \quad i = 1, 28 \\ \pi & \text{for } \phi_i \quad i = 1, 8 \end{cases}$$

and

d = arbitrary divisor (taken as 4 initially).

Addition of the corresponding random step to the previous values of λ, θ, and ϕ results in another value of $\underline{P}$. If $\underline{P} > P^*$, a step in the opposite direction is taken by setting $\Delta u = -\Delta u$, the consistency with the constraints on λ, θ, and ϕ is checked again, and $\underline{P}$ is recalculated. If $\underline{P}$ is still greater than $\underline{P}^*$, a new random step is chosen and added to the point associated with $\underline{P}^*$. As long as $\underline{P} < \underline{P}^*$, we set the new value of $\underline{P}^* = \underline{P}$ and the step size is doubled until $\underline{P} > \underline{P}^*$, then a new random step is instituted from the point (λ, θ, ϕ) associated with $\underline{P}^*$.

The accelerated random search is based on the concept of randomly choosing a search direction and a step size, and searching in that direction until a minimum is found. At the minimum, a new random direction and step size are chosen and the process is repeated. As the search gets closer to the minimum, i.e., $\underline{P}^*$ decreases, the variance of the distribution from which the random step is drawn is decreased to facilitate accurate determination of the minimum.

The last change that was incorporated into the algorithm was a "creeping aspect" of the random search by which the random steps (Δu's) become either larger or smaller as required. In particular, if $d = 4$ in the expression for Δu, as prescribed above, and say 100 directional steps are looked at with no improvement, then d is halved, thereby doubling Δu. Another one hundred trials with no improvement cause another halving of d, etc. If an improvement is obtained, d is set back to 4 and the expansion begins anew. If after, say, a total of 500 such trials where u is 16 times its original value and no improvement has been found, then d is set back to 4 and is consecutively doubled to decrease the step size in the same manner as the step size was increased above. Thus the creeping random search has an expanding and contracting facility, which proved useful in determining the best vol^{-1} estimate.

APPENDIX E

INTERPRETATION OF NUMERICAL RESULTS

This appendix describes explicitly the method used for computing approximate numerical bounds on the physical variables corresponding to a given volume estimate. Let the volume estimate

$$J(Q) = [\ell^n / \det P]^{\frac{1}{2}} \tag{E-1}$$

result from the matrix Q, the corresponding positive definite matrix

$$P = C\Lambda C^T \tag{E-2}$$

where $\Lambda = \text{diag}\ (\lambda_1, \ldots, \lambda_n)$ is the diagonal matrix containing the eigenvalues of P, the columns of the orthogonal matrix C are the (normalized) eigenvectors of P, and the quadratic volume estimate

$$x^T P x < \ell \ . \tag{E-3}$$

Putting

$$y = Cx \tag{E-4}$$

transforms (E-3) into the diagonal form

$$y^T \Lambda y < \ell \ . \tag{E-5}$$

The maximum intercept of $y^T \Lambda y = \ell$ with the y_i axis is at the point

$$y_i^m = (\ell / \lambda_i)^{\frac{1}{2}} \ . \tag{E-6}$$

Let C_i^T denote the transpose of row i of the matrix C and let

$$x^{(i)} = y_i^m C_i \ .$$

Then the entries of $x^{(i)}$ provide conservative estimates of the maximum values of the physical variables; the vectors $x_i (i = 1, \ldots, n)$

are in fact the points in x-space at the extremities of the principal axes of the hyperellipse $x^T P x = \ell$.

For example, consider the best run (number 6) at the Institute for Space Studies with

$$\text{vol}^{-1} = 0.488 \times 10^{35} , \qquad \text{(E-7)}$$

$$Q = \begin{bmatrix} 4.16 & -5.82 & 3.68 & 0 & 0 & 0 & 0 & 0 & 0 \\ -5.82 & 8.37 & -5.17 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3.68 & -5.17 & 3.26 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.068 & -0.122 & 0.311 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.122 & 2.16 & -0.537 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.311 & -0.537 & 1.42 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.074 & 0.085 & 0.340 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.085 & 1.14 & 0.400 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.340 & 0.400 & 1.57 \end{bmatrix} ,$$

(E-8)

$$\ell = 9.604 \times 10^{-9} , \qquad \text{(E-9)}$$

$$\Lambda = \begin{bmatrix} 208.1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 32.7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 18.8 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2.12 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.428 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.403 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.00800 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.00313 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.00142 \end{bmatrix} ,$$

(E-10)

$$C = \begin{bmatrix} 0.006 & 0 & 0 & 0.751 & 0 & 0 & 0.660 & 0 & 0 \\ 1.00 & 0 & 0 & -0.008 & 0 & 0 & 0.008 & 0 & 0 \\ 0.011 & 0 & 0 & 0.660 & 0 & 0 & -0.751 & 0 & 0 \\ 0 & 0.001 & 0 & 0 & 0 & -0.223 & 0 & -0.975 & 0 \\ 0 & 1.00 & 0 & 0 & 0 & 0.031 & 0 & -0.006 & 0 \\ 0 & 0.032 & 0 & 0 & 0 & -0.974 & 0 & 0.223 & 0 \\ 0 & 0 & -0.001 & 0 & -2.24 & 0 & 0 & 0 & -0.975 \\ 0 & 0 & -1.00 & 0 & 0.035 & 0 & 0 & 0 & -0.007 \\ 0 & 0 & -0.035 & 0 & -0.974 & 0 & 0 & 0 & 0.224 \end{bmatrix}, \tag{E-11}$$

and corresponding vectors $x^{(1)}, \ldots, x^{(9)}$ given by

$$\begin{bmatrix} .39 \times 10^{-8} \\ .68 \times 10^{-5} \\ .82 \times 10^{-7} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ .25 \times 10^{-7} \\ 1.7 \times 10^{-5} \\ .54 \times 10^{-6} \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -.32 \times 10^{-7} \\ -2.3 \times 10^{-5} \\ .81 \times 10^{-6} \end{bmatrix}, \begin{bmatrix} .51 \times 10^{-4} \\ -.54 \times 10^{-6} \\ .45 \times 10^{-4} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -.33 \times 10^{-4} \\ .51 \times 10^{-5} \\ -1.5 \times 10^{-4} \end{bmatrix},$$

(E-12)

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ -.33 \times 10^{-4} \\ .47 \times 10^{-5} \\ -1.5 \times 10^{-4} \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} .73 \times 10^{-3} \\ .91 \times 10^{-5} \\ -.83 \times 10^{-3} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ -1.7 \times 10^{-3} \\ -1.0 \times 10^{-5} \\ 0.4 \times 10^{-3} \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -2.5 \times 10^{-3} \\ -1.7 \times 10^{-5} \\ .57 \times 10^{-3} \end{bmatrix}.$$

Finding the maximum first entry in these vectors yields 0.73×10^{-3} radians = 2.48 min as an estimate for the maximum absolute value of $x_1 = \phi$ from which the physical system will settle. The variables $x_4 = \theta$ and $x_7 = \psi$ are handled just as x_1. The variables $x_2 = v''_\phi$, $x_3 = \omega''_\phi$, $x_5 = v''_\theta$, $x_6 = \omega''_\theta$, $x_8 = v''_p$, and $x_9 = \omega''_\psi$ are estimated in the same way, but the results are dimensionless so that transformation to physical variables requires further scaling by use of (A-48), namely

$$\begin{aligned} v'_i &= K_m K_c v''_i \qquad i = \phi, \theta, \psi \\ \omega'_i &= \frac{K_c \tau_1}{\tau_2} \omega''_i \qquad i = \phi, \theta, \psi \end{aligned} \tag{E-13}$$

and then removing the zero equilibrium by use of (A-37) (note that $\phi_e = \theta_e = \psi_e = 0$) namely

$$\begin{aligned} v_i &= v'_i + I h^o_i \qquad i = \phi, \theta, \psi \\ \omega_i &= \omega'_i - \frac{\tau_1 I}{\tau_2 K_m} h^o_i \qquad i = \phi, \theta, \psi \end{aligned} \tag{E-14}$$

APPENDIX F

OTHER COMPUTATIONAL TECHNIQUES

(i) Algorithm Based on Selecting P Directly

Computer runs were made using a technique of generating a positive definite P-matrix directly, without generating Q and then solving the Liapunov equation

$$PA + A^T P = - Q \qquad (F\text{-}1)$$

for P. This work originated with the motive of speeding the routine so that a great many matrices could be examined with a minimum of machine time. The abort feature described previously was essential to this plan, because the fact that P is positive definite does not ensure the definiteness of Q. Use of the abort procedure without direct generation of P had reduced the time required to investigate a matrix from 30 seconds to an average of about 10 seconds (2 seconds to search nine dimensional space and 8 seconds to generate Q and solve the Liapunov equation for P). The time saved was not the only advantage of direct P-matrix generation.

It soon became clear that direct generation of the P-matrix has other advantages. The eigenvalues and eigenvectors of P have direct physical interpretation in the nine dimensional state space, so that picking P directly permits more insight into the geometry of parameter space than the Q-generation procedure. In addition, when the P-matrix is generated directly, the A-matrix is unnecessary, and the time derivative $\dot{x}$ can be efficiently computed directly from the nonlinear function. By setting $x_1 = \phi'$ $x_2 = v''_\phi$, $x_3 = \omega''_\phi$, $x_4 = \theta'$, $x_5 = v''_\theta$, $x_6 = \omega''_\theta$, $x_7 = \psi'$, $x_8 = v''_\psi$, $x_9 = \omega''_\psi$, the equations of Fig. A-8 may be written in the form (for all initial momenta zero):

$$\begin{aligned}
\dot{x}_1 &= - a x_2 - a(x_5 \sin x_1 + x_8 \cos x_1) \tan x_4 \\
\dot{x}_2 &= - b x_2 + \frac{b}{k} \operatorname{sat}(10k\Delta\gamma_1 + 9kx_3) \qquad (F\text{-}2) \\
\dot{x}_3 &= - 2(x_3 + \Delta\gamma_1)
\end{aligned}$$

$$
\begin{aligned}
\dot{x}_4 &= -a(x_5 \cos x_1 - x_8 \sin x_1) \\
\dot{x}_5 &= -bx_5 + \frac{b}{k}\,\mathrm{sat}\left(10d_{12}k(\Delta\beta_1 \cos \Gamma_2 + \Delta\beta_2 \cos \Gamma_1 + \frac{9}{2}x_6)\right) \\
\dot{x}_6 &= -2x_6 - 2d_{12}(\cos \Gamma_2\Delta\beta_1 + \cos \Gamma_1\Delta\beta_2) \\
\dot{x}_7 &= -a(x_5 \sin x_1 + x_8 \cos x_1)/\cos x_4 \\
\dot{x}_8 &= -b(x_8 - \mathrm{sat}\left(10d_{12}k(-\Delta\beta_1 \sin \Gamma_2 - \Delta\beta_2 \sin \Gamma_1 + \frac{9}{2}x_9)\right) \\
\dot{x}_9 &= -2x_9 + 2d_{12}(\sin \Gamma_2\Delta\beta_1 + \sin \Gamma_1\Delta\beta_2)\ ,
\end{aligned}
\qquad \text{(F-2) (Cont.)}
$$

where

$$
a = K_mK_c/I,\quad b = \frac{1}{\tau_m},\quad k = K_c,\quad d_{12} = 2\ \mathrm{sgn}(\gamma_{1c} - \gamma_{2c})\ ,
$$
$$
h_\theta^o = h_\phi^o = h_\psi^o = 0,\quad \Gamma_1 = \Delta\gamma_1 + \gamma_{1c},\quad \Gamma_2 = \Delta\gamma_2 + \gamma_{2c} \qquad \text{(F-3)}
$$

relates the present notation to that of Fig. A-8 and $\mathrm{sat}(u) = +26$ for $26 < u$, $\mathrm{sat}(u) = u$ for $+26 \geq u \geq -26$, $\mathrm{sat}(u) = -26$ for $u < -26$. The main improvement in speed is in the calculation of $\Delta\gamma_i$ and $\Delta\beta_i$. Specifically, by using the upper sign for $i = 1$ and the lower for $i = 2$ in (F-5),

$$
\Delta\gamma_i = \tan^{-1}\left(\frac{\epsilon_\alpha - \epsilon_\beta \tan \gamma_{ic}}{1 + \epsilon_\beta(\tan \gamma_{ic} + \epsilon_\alpha)\tan \gamma_{ic}}\right)\ , \qquad \text{(F-4)}
$$

where

$$
\begin{aligned}
\epsilon_\alpha &= (\cos x_4 - 1)\tan \gamma_{ic} \mp \sin x_7 \tan x_1 \frac{\tan \beta_{ic}}{\cos \gamma_{ic}} \\
&\quad \mp \cos x_7 \sin x_4 \frac{\tan \beta_{ic}}{\cos \gamma_{ic}} + \cos x_7 \tan x_1 - \sin x_7 \sin x_4\ , \\
\epsilon_\beta &= (\cos x_7 - 1) + \sin x_7 \sin x_4 \tan x_1 \mp \sin x_7 \frac{\tan \beta_{ic}}{\cos \gamma_{ic}} \\
&\quad \pm \tan x_1 \sin x_4 \cos x_7 \frac{\tan \beta_{ic}}{\cos \gamma_{ic}} - \tan x_1 \cos x_4 \tan \gamma_{ic}\ .
\end{aligned}
\qquad \text{(F-5)}
$$

The formula in Fig. A-8 is used to compute $\Delta\beta_i$ unless round-off error would be significant because $\Delta\beta_i < 0.1$, in which case $\Delta\beta = \sin^{-1}\mu$, where μ is the iteratively obtained solution of

$$\kappa = -a \tan \beta_{ic} + \mu , \qquad \text{(F-6)}$$

in which

$$1 - a = \cos \Delta\beta_i = \sqrt{1 - \mu^2} = 1 - \tfrac{1}{2}\mu^2 - \frac{1}{2\cdot 4}\mu^4 - \frac{1\cdot 1\cdot 3}{2\cdot 4\cdot 6}\mu^6 , \qquad \text{(F-7)}$$

and letting $b_j = \cos x_j - 1$, $j = 4,7$,

$$\kappa = (b_7 + b_4 + b_7 b_4)\tan \beta_{ic} \pm \sin x_7 (1 + b_4)\cos \gamma_{ic} \pm \sin x_4 \sin \gamma_{ic} , \qquad \text{(F-8)}$$

where the upper sign is used for $i = 1$ and the lower sign for $i = 2$. The procedure is fast, and does not require double precision arithmetic.

Initial experiments with direct generation of the P-matrix proceeded by using the classical Gershgorin theorem [31] and applying the results to the OAO.

The methods shown in (F-2) through (F-8), when programmed for an IBM 360/75, investigates over 50,000 P matrices per hour. Additional compactness and speed are obtained by generation of points x by letting $x = Cy$ as in (D-74), where

$$y_i = \xi_i R \Big/ \Big(\sum_{j=1}^{n} \xi_i^2 \Big)^{\frac{1}{2}} , \qquad \text{(F-9)}$$

in which R, ξ_1, ..., ξ_n are independent, each ξ_i is uniformly distributed on $(-1, +1)$, and R is so distributed that $\text{Prob}(R < r) = r^n$. This procedure generates uniformly (by volume) distributed random points y without discarding points as described following Fig. D-2.

Initial experiments generated 35,000 positive definite matrices P in about 15 minutes of computer time, revealing no positive definite matrices Q, i.e., the estimate of the domain was $J(P) = 0$ for each P. The suspected reason is that relatively

few positive definite matrices P have associated positive definite matrices Q. The best positive definite matrix from the Q-generation method was therefore factored and used to generate a starting P matrix for this method, with inconclusive results.

This procedure shows promise at the present writing, but the investigators were unable to refine it further during the present contract.

(ii) On Improved Estimates via Luré-Liapunov Functions

In developing the optimal quadratic estimate of the domain of attraction we chose to use a quadratic form Liapunov function because the resulting estimate is easy to visualize and interpret, it compares favorably with more complex estimates, and because it eliminates a number of computational problems. However, implicit in this choice is the assumption that a sufficiently large family of ellipsoids can be found such that none of the system trajectories leave any member of the family once they are interior to that member. This may be a severe restraint considering that the system nonlinearities are not accounted for in the generic shape of the estimate. This section describes a possible method for accounting for some of the nonlinear effects in the generic shape of the estimate.

Consider the fact that the system model can be rewritten to identify the "dominant" nonlinearity as follows:

$$\begin{aligned} \dot{x} &= Ax + Bf^a(\sigma) + \epsilon \left(g(x) - Bf^a(\sigma) \right) \\ \sigma &= Cx \end{aligned} \qquad \text{(F-10)}$$

where A is as before, $f^a(x)$ is the 3-vector of motor saturation functions with linearized 3-vector argument σ, B and C are gain matrices, the term in parentheses is the new collection of nonlinearities and ϵ is a perturbation parameter. Note that for $\epsilon = 1$ we have the original system model, and for $\epsilon = 0$ we have the Popov approximation to the system model.

Now consider the Luré-Liapunov function

$$V(x, \sigma) = x^T P x + \int_0^{\sigma} \left(f^a(\sigma) \right)^T d\sigma , \qquad P > 0 \qquad \text{(F-11)}$$

and its derivative

$$\dot{V}(x, \sigma) = - x^T Q x + x^T (2PB + A^T) f^a(\sigma) + \left(f^a(\sigma)\right)^T B f^a(\sigma) + \epsilon \left[\left[2x^T P + \left(f^a(\sigma)\right)^T \right] \left[g(x) - B f^a(\sigma) \right] \right] \tag{F-12}$$

The function $V(x, \sigma)$ is positive definite since the integral of the saturation function vector is positive $\left(f_i^a(\sigma) = f_i^a(\sigma_i)\right)$. The relationship between P and Q is still given by the Liapunov equation. It is required to show that $\dot{V}(x, \sigma)$ is negative over at least some finite neighborhood of the state space origin so that LaSalle's theorem (Section 3) can be applied to obtain an estimate.

Before we investigate this further, let us note the computational complexities introduced by this new formulation. First, unless the quadratic form strongly dominates the integral term in $V(x, \sigma)$ the surface $V(x, \sigma) = C$ will not be like an ellipsoid. Therefore the search procedure for the minimum of $V(x, \sigma)$ on $\dot{V}(x, \sigma) = 0$ must be revised since it is based on putting a box around the ellipsoid. Secondly, since the estimate is no longer an ellipsoid the calculation of its volume is much more complicated. Thus, before launching a substantial effort to solve these problems one would like some assurance that the new procedure will be an improvement. This can be obtained if the function $\dot{V}(x, \sigma)$ is negative definite for $\epsilon = 0$, i.e., if the Popov approximation is globally asymptotically stable.

Kalman [30] has proven the equivalence of Luré-Liapunov functions and Popov's condition, i.e., a Luré-Liapunov function of the form given above can be constructed with negative definite derivative if the Popov theory shows the system to be absolutely stable. Unfortunately, as stated in Appendix C, this cannot be done with the theory as it now stands. Thus, this approach was not carried any further in view of our inability to justify the effort that would be required to solve the outstanding computational problems indicated above.

APPENDIX G

DETAILED FLOW CHARTS

This appendix contains flow charts of the stability analysis algorithms where a Q-matrix is initially selected and where a P-matrix is initially selected. Both algorithms are basically the same, the primary difference being that in the Q-matrix selection an inversion process is required to determine the associated P-matrix (which is assuredly positive definite), whereas in the P-matrix selection the resulting Q-matrix (not necessarily positive definite) found by a matrix multiplication must be tested to ascertain its character. All Q-matrices that result from picking a P-matrix must be discarded if they prove to be semidefinite or negative definite because of the theory being utilized.

A thumbnail sketch of each of the subroutines shown in the flow charts (Figs. G-1 and G-2) follows.

Subroutine AFX

This subroutine calculates the matrix A and the nonlinear vector $g(x)$ of the equations of motion $\dot{x} = Ax + g(x)$.

Subroutine QGEN

This subroutine generates a positive definite matrix Q given a set of $n(n + 1)/2$ independent variables as given in Appendix D part (ii).

Subroutine DSRCH

This subroutine is the random search subroutine for the state space where a min V with $\dot{V} = 0$ is to be achieved.

Subroutine PEAIQ

This subroutine solves the equation $A^TP + PA = -Q$ for a positive definite P-matrix, given a stable A-matrix and a positive definite Q-matrix.

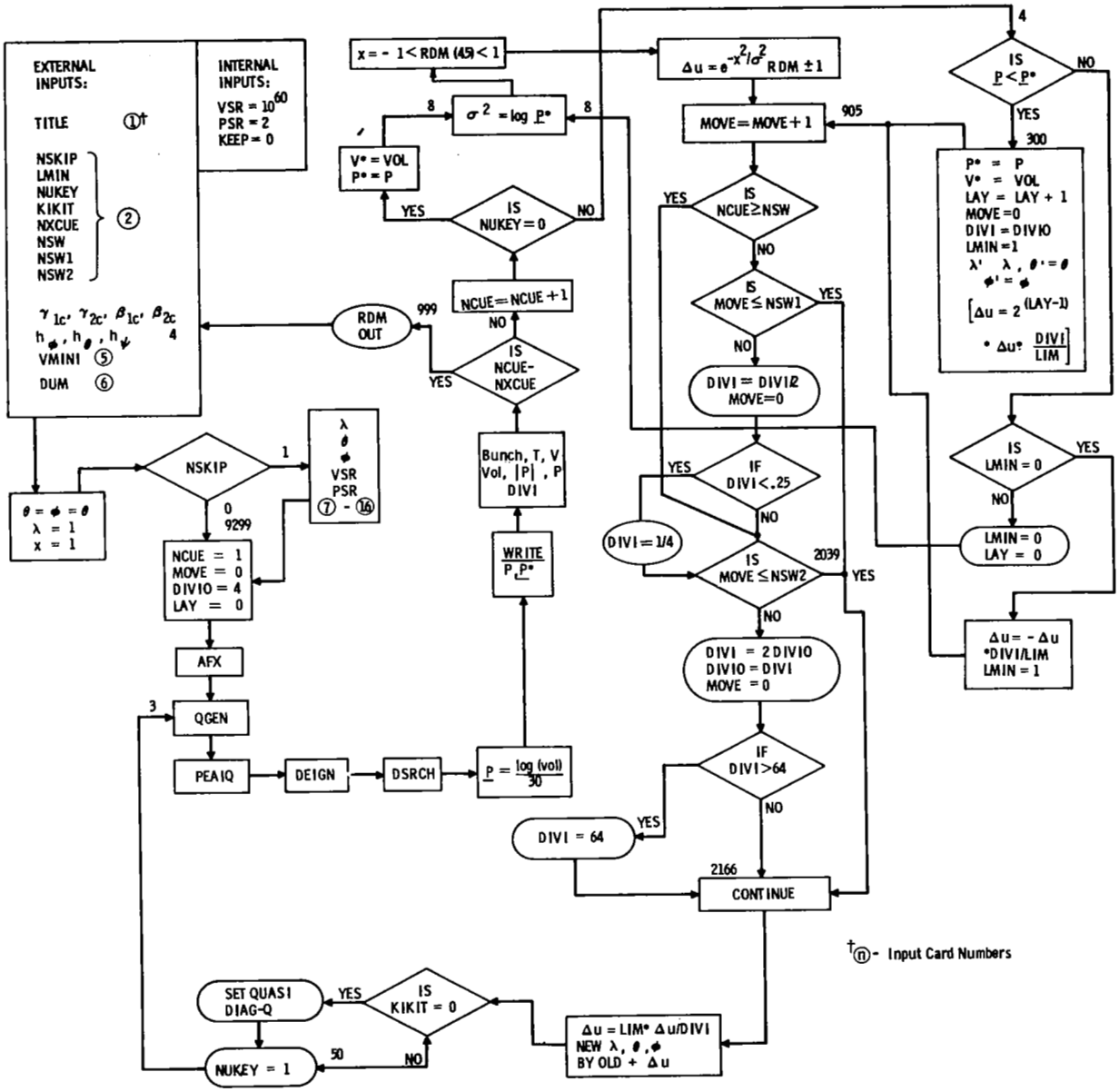


Fig. G-1 Flow Chart for Algorithm Based on Q-Matrix

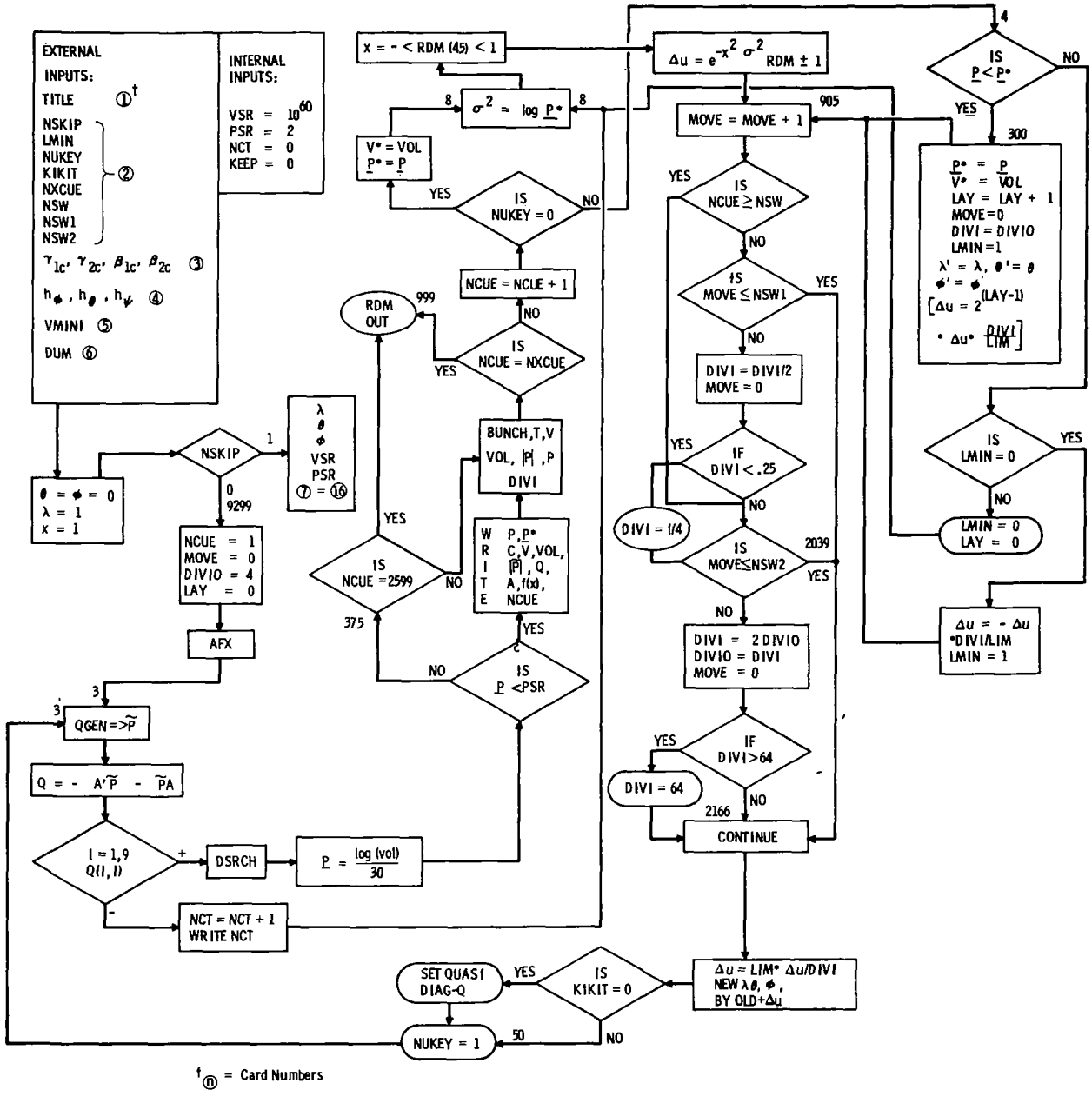


Fig. G-2 Flow Chart for Algorithm Based on P-Matrix

Subroutine DEIGN

This subroutine calculates the eigenvalues and eigenvectors of the positive definite P-matrix.

The logic of the Q-parameter or P-parameter search is included in the flow charts G-1 and G-2, respectively, since this directional random search procedure is applicable to most any system.

Figure G-3 is a flow chart of the simulation of the system where in most of the subroutines listed above were utilized and augmented as listed below.

Subroutine DERIV

This subroutine is the set of equations of motion of the system $\dot{x} = Ax + g(x)$.

Subroutine JINPG

This subroutine contains a fourth order Runge-Kutta integration scheme with fixed step size.

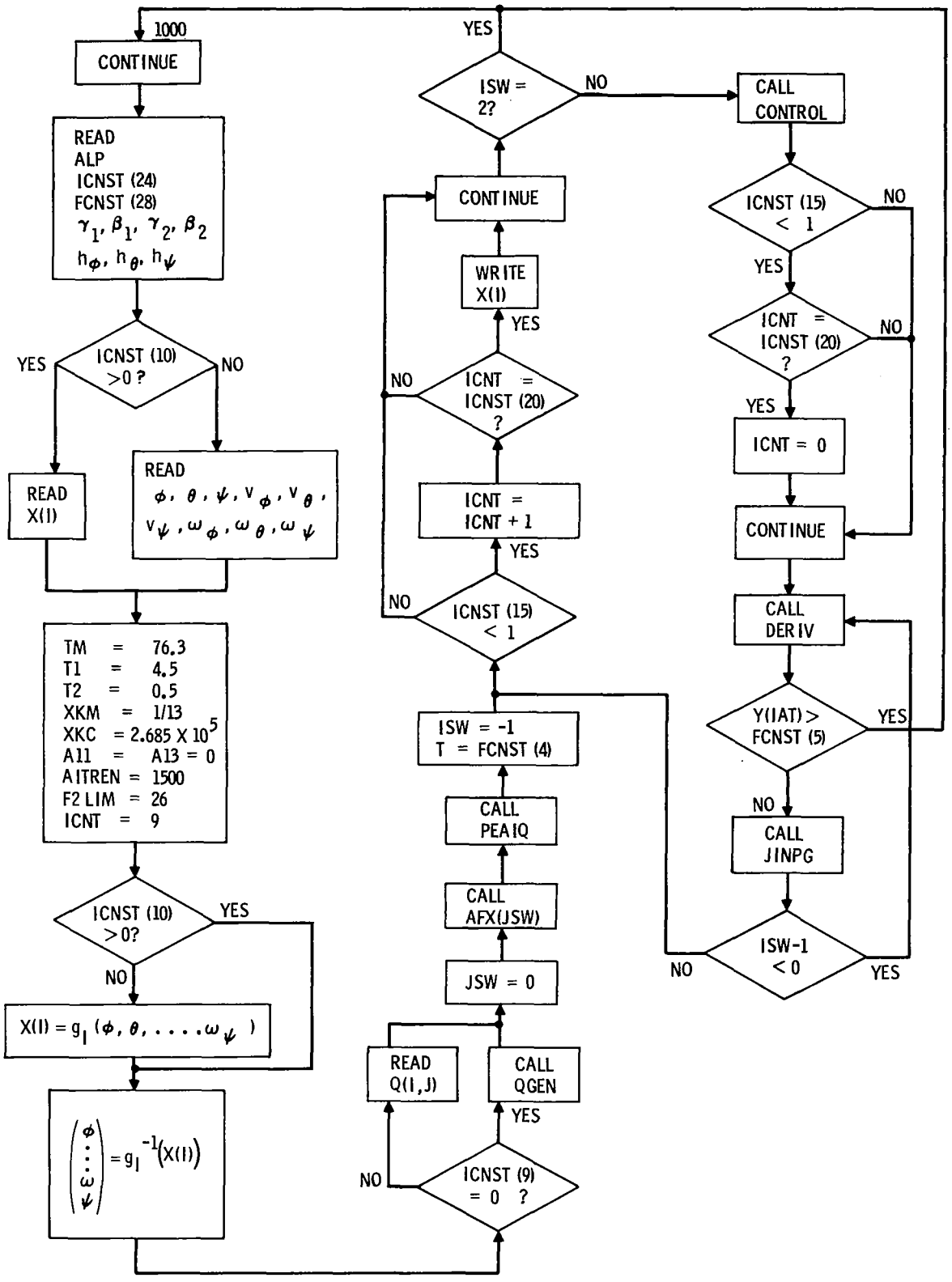


Fig. G-3 Generic Simulation Program Flow Chart

REFERENCES

1. Doolin, B. and R. D. Showman, "Attitude Stabilization of Satellites with Gimbaled Star Trackers," presented at Second IFAC Symposium on Automatic Control in Space, Vienna, Austria, September 1967.

2. Showman, R. D., "Simplified Processing of Star Tracker Commands for Satellite Attitude Control," IEEE Transactions on Automatic Control, Vol. AC-12, No. 4, August 1967, pp. 353-359.

3. Geiss, G. R., The Analysis and Design of Nonlinear Control Systems via Liapunov's Direct Method, Ph.D. dissertation, Polytechnic Institute of Brooklyn, Brooklyn, New York, June 1964.

4. Margolis, S. G. and W. G. Vogt, "Control Engineering Applications of V. I. Zubov's Construction Procedure for Liapunov Functions," IEEE Trans. on Automatic Control, Vol. AC-8, No. 2, pp. 104-113, April 1963.

5. Geiss, G. R., "Estimation of the Domain of Attraction," Progress Report No. 8 - Studies in the Fields of Space Flight and Guidance Theory, NASA George C. Marshall Space Flight Center, Huntsville, Alabama, June 1966.

6. LaSalle, J. P. and S. Lefschetz, Stability by Liapunov's Direct Method with Applications, Academic Press, New York, 1961.

7. Rodden, J. J., Applications of Liapunov Stability Theory, Ph.D. dissertation, Stanford University, 1964; also Lockheed Missile and Space Company, LMSC/A603476, April 1964.

8. Weissenberger, S., Stability-Boundary Approximation for Relay-Control Systems via the Direct Method of Liapunov, Ph.D. dissertation, Stanford University, 1964, NASA CR-320, October 1965.

9. Murnaghan, F. D., Lectures on Applied Mathematics, Vol. III: The Unitary and Rotation Groups, Spartan Books, Washington, D.C., 1962.

10. Barron, R. L., "Inference of Vehicle and Atmosphere Parameters from Free Flight Motions," presented at AIAA Guidance, Control and Flight Dynamics Conference, Huntsville, Alabama, August 14-16, 1967, AIAA Paper No. 67-600.

11. Geiss, G. R., "The Effect of Star Tracker Model Choice on Satellite Attitude Stability Analysis," IEEE Transactions on Automatic Control, Vol. AC-14, No. 6, December 1969.

12. Rothschild, D. and Jameson, A., "Comparison of Four Numerical Algorithms for Solving the Liapunov Matrix Equation," Int. J. Control, Vol. 11, 1970, pp. 181-198.

13. Papsco, R. E., "The Stabilization and Control System for the Orbiting Astronomical Observatory," Navigation - Journal of the Institute of Navigation, Vol. 11, No. 1, Spring 1964.

14. Zetkov, G. and R. Fleissig, "Dynamic Analysis of OAO Spacecraft Motion by Analog-Digital Simulation," IRE International Convention, New York, 1962.

15. Aizerman, M. A. and F. R. Gantmacher, Absolute Stability of Regulator Systems, transl. E. Polak, Holden-Day, Inc., San Francisco, 1964, p. 52.

16. Brockett, R. W., "The Status of Stability Theory for Deterministic Systems," IEEE Intl. Conv. Record, Pt. 6, 1966, pp. 125-142.

17. Sandberg, I. W., "On the Boundedness of Solutions of Nonlinear Integral Equations," Bell System Technical Journal, Vol. 44, pp. 439-453, 1965.

18. Yacubovich, V. A., "Frequency Conditions for the Absolute Stability of Control Systems with Several Nonlinear or Linear Nonstationary Blocks," Automation and Remote Control, N. 6, 1967, pp. 857-880.

19. Geiss, G. R., V. Cohen, et al., Development of an Algorithm for the Nonlinear Stability Analysis of the Orbiting Astronomical Observatory Control System, Grumman Research Department, RE-307, November 1967.

20. Ma, Er-Chieh, "A Finite Series Solution of the Matrix Equation $Ax-xB = C$," Journal of SIAM on Applied Mathematics, Vol. 11, No. 3, May 1966.

21. Smith, R. A., "Matrix Calculations for Liapunov Quadratic Forms," Journal of Differential Equations, Vol. 2, No. 2, April 1966.

22. Jameson, A., "Solution of the Equation $AX + XB = C$ by Inversion of an $n \times n$, or $m \times m$ Matrix," SIAM Journal on Applied Mathematics, Vol. 16, No. 5, September 1968.

23. Thrall, R. M. and Tornheim, L., Vector Spaces and Matrices, John Wiley & Sons, New York, 1957, p. 67.

24. Peterson, W., Error-Correcting Codes, the M.I.T. Press and J. Wiley & Sons, New York, 1962.

25. Bocher, M., Introduction to Higher Algebra, Macmillan, New York, 1947.

26. Gantmacher, F. R., The Theory of Matrices, Vol. 1, Chelsea Publishing Company, New York, 1959, p. 43.

27. Taylor, G. E., MIN-ALL: An Iterative Procedure for Obtaining Local Minimums, Grumman Research Department Memorandum RM-374, August 1967.

28. Davidon, W. C., Variable Metric Method for Minimization, Atomic Energy Commission Research and Development Report ANL-5990 (Rev.), 1959.

29. Fletcher, R. and M. J. D. Powell, "A Rapidly Convergent Descent Method for Minimization," The Computer Journal, Vol. 6, 1963, pp. 163-168.

30. Kalman, R. E., "Liapunov Functions for the Problem of Luré in Automatic Control," Proc. Natl. Acad. of Sci., Vol. 49, No. 2, February, 1963, pp. 201-205.

31. Householder, A. S., "Norms and the Localization of Roots of Matrices," Bull. Amer. Math. Soc., Vol. 74, 1968, p. 822.